

ESD ACCESSION LIST

ESTI Call No. 70367

Copy No. 1 of 2 c/s

ESD-TR-70-204

M69-65

ANNOTATED MICROPROGRAMMING BIBLIOGRAPHY

Microprogramming Group

JULY 1970

Prepared for

DIRECTORATE OF PLANNING AND TECHNOLOGY

ELECTRONIC SYSTEMS DIVISION

AIR FORCE SYSTEMS COMMAND

UNITED STATES AIR FORCE

L. G. Hanscom Field, Bedford, Massachusetts

ESD RECORD COPY

RETURN TO
SCIENTIFIC & TECHNICAL INFORMATION DIVISION
(ESTI), BUILDING 1211



This document has been approved for public release and sale; its distribution is unlimited.

Project 700A

Prepared by

THE MITRE CORPORATION

Bedford, Massachusetts

Contract F19(628)-68-C-0365

AD0709765

When U.S. Government drawings, specifications, or other data are used for any purpose other than a definitely related government procurement operation, the government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Do not return this copy. Retain or destroy.

ANNOTATED MICROPROGRAMMING BIBLIOGRAPHY

Microprogramming Group

JULY 1970

Prepared for

DIRECTORATE OF PLANNING AND TECHNOLOGY

ELECTRONIC SYSTEMS DIVISION

AIR FORCE SYSTEMS COMMAND

UNITED STATES AIR FORCE

L. G. Hanscom Field, Bedford, Massachusetts



This document has been approved for public release and sale; its distribution is unlimited.

Project 700A

Prepared by

THE MITRE CORPORATION

Bedford, Massachusetts

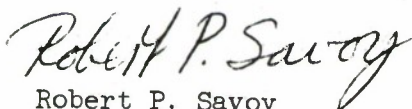
Contract F19(628)-68-C-0365

FOREWORD

This report was prepared by The MITRE Corporation, Bedford, Massachusetts, under Air Force Contract No. F19(628)-68-C-0365. It carries MITRE Project No. 700A. There is no Air Force System, Project or Task Number assigned. The report is a bibliography which provides an indexed, annotated list of publications related to the field of microprogramming. It covers the period from 1951, when the term was introduced, until the present time.

REVIEW AND APPROVAL

This technical report has been reviewed and is approved.



Robert P. Savoy
Electronics Engineer
Development Engineering Division
Directorate of Planning & Technology

ABSTRACT

This bibliography provides an indexed, annotated list of publications related to the field of microprogramming, from the introduction of the term in 1951 to the present time.

ACKNOWLEDGEMENT

Documents for this bibliography were collected by Thomas Berschback and Nancy Anschuetz. A number of entries were obtained from the bibliography published by John Douglas, General Electric, Phoenix, in the SICMICRO Newsletter, Vol. 1, No. 2.

After initial review by Thomas Berschback, revisions were made by Robert W. Cornelli, Robert G. Curtis, Barbara J. Huberman and Codie S. Wells.

The KWIC index program used was prepared by William Amory.

Editorial control was exercised by Thomas Berschback throughout most of the preparation process.

TABLE OF CONTENTS

	<u>Page</u>
SECTION I INTRODUCTION	1
SECTION II BIBLIOGRAPHY	3
SECTION III INDEX	51

SECTION I

INTRODUCTION

This bibliography provides an indexed, annotated list of publications related to the field of microprogramming, from the introduction of the term in 1951 to the present time.

This bibliography has been winnowed to exclude publications related only to the technological aspects of the memories, whether read-only or read-write, used to implement microprograms.

Section II is the annotated bibliography. Entries are organized alphabetically, by first author. Otherwise unidentified quoted sections are taken from the paper.

Section III contains a computer generated index, which provides access to bibliography entries by:

1. Key words in titles.
2. Authors' names.
3. Date of publication.

SECTION II

BIBLIOGRAPHY

(1)

ALLEN-BABCOCK COMPUTING, INC., "Information outline: Remote-access time-sharing system".

This sales brochure for ABC briefly describes their time-sharing system. Their 360/50 "...has been modified by the attachment of a second Read-only-store (ROS) which contains additional operation codes...performs operations such as list searching, evaluation of PL/I expressions, floating decimal arithmetic, and various utility functions, all of which result in the fastest interpretive system in existence."

(2)

ALLEN, M. W.; PEARCEY, T.; PENNY, J. P.; ROSE, G. A. and SANDERSON, J. G., "CIRRUS, An economical multiprogram computer with microprogram control", *IEEE Transactions on Electronic Computers*, Vol. EC-12; December 1963, pp. 663-671.

Machine: CIRRUS

CIRRUS, constructed in 1963 at the University of Adelaide, Australia, is a general-purpose computer built for test purposes and is not being produced. Some mention is made of microprogramming, but only as a mechanism which allows certain operations to be handled with more efficiency.

(3)

AMDAHL, L. D., "Microprogramming and stored logic", *Datamation*, Vol. 10, No. 2; February 1964, pp. 24-26.

Machines: Thompson Ramo Wooldridge 133 (now the Bunker-Ramo 133),
Packard-Bell 440, Collins 8401

"Microprogrammed computers basically differ from conventional computers in that the equivalent of a single instruction in a conventional computer is often a subroutine (interpreted instruction) in a microprogrammed computer." In such computers a program can be considered a calling sequence. The calling sequences of the machines listed above are described to show how the control sequences are simplified and more flexibility is allowed through the use of microprogramming. The cost of additional execution time is compared with the flexibility provided by microprogramming and the time saved in executing some complex instructions.

(4)

BAILEY, S. J., "Faster computer control with a read-only memory", *Control Engineering*, Vol. 14, No. 8; August 1967, pp. 65-68.

Machine: Interdata 3

This article presents several applications of Interdata 3 computers with special microprograms implemented to achieve significantly improved efficiency. The author claims that the use of read-only memories makes the use of small computers (like the Interdata 3) practical for application to process control problems.

(5)

BASHKOW, T. R.; SASSON, A. and KRONFELD, A., "System design of a FORTRAN machine", *IEEE Transactions on Electronic Computers*, Vol. EC-16, No. 4; August 1967, pp. 485-499.

"The authors suggest an approach to the use of algebraic languages, by means of hardware interpreter. A design is presented for the construction of a machine for the execution of a subset of FORTRAN consisting of arithmetic assignment, GO TO, computed GO TO, arithmetic IF, PAUSE, DO, CONTINUE, unformatted READ and WRITE, END and DIMENSION statements. Although such a machine has not been constructed, the authors make it clear that such construction is feasible. State diagrams are given for some of the most important circuits, and the hardware implementation is given for one of these circuits."

Computing Reviews; January 1968,
#13,542, p. 63.

(6)

BAZERQUE, G.; FERRIE, J. and HUGOT, P., "Micromachine universal structure study oriented to computers simulation", *Preprints of papers presented at IFIP Congress 68, Edinburgh; Booklet D: Hardware I*, pp. D103-D107.

Machine: IBM 360/30

This article, originally in French and only roughly translated, addresses the design of a general-purpose or universal system to simulate any microprogrammed computer. As an example, a simulation of an IBM 360/30 is presented - a convenient choice because both the micromachine and the 360/30 contain 32 bit words. The authors note that the quality of the simulation is greatly enhanced by such matching where possible. The simulation of the 360/30 includes representations of general registers, instruction location counter and core memory, but not the I/O operations.

(7)

BEELITZ, H. R.; LEVY, S. Y.; LINHARDT, R. J. and MILLER, H. S., "System architecture for large-scale integration", *AFIPS Conference Proceedings*; FJCC 1967, pp. 185-200.

Machine: LIMAC

This article describes an approach for simplifying the control structure of a machine which utilizes the benefits of large-scale integration (LSI). "This evolutionary development has gone from conventional machine design first to a one-dimensional functional control partitioning (the register machine) and then to a two-dimensional elementary operation formatted control partitioning (the LIMAC machine)."

The two "dimensions" are: information transfer and data processing execution. Use of this structure "...leads naturally to the elementary operation (EO) format of control (also known as microprogramming)." In LIMAC (Large Integrated Monolithic Array Computer), a machine instruction is executed by performing a sequence of microinstructions. First, the operands are transferred into a function module, then another transfer operation passes the control word into a register to specify the execution.

(8)

BECK, L. and KEELER, F., "The C-8401 data processor", *Datamation*, Vol. 5, No. 2; February 1964, pp. 33-35.

Machine: The Collins Radio C-8401

The C-8401 is "...a stored program, intermediate scale digital computer...a microprogrammed data processor...communications oriented." The 1024 word, 36 bit micro-memory, though non-destructive, "...may be conveniently loaded under operator or program control."

(9)

BENJAMIN, R. I., "The Spectra 70/45 emulator for the RCA 301", *Communications of the ACM*, Vol. 8, No. 12; December 1965, pp. 748-752.

Machine: RCA Spectra 70/45

The author, with startling brevity, discusses the concept of emulation, describes the RCA 301, the RCA Spectra 70/45 and its micromachine and the emulation process as applied to emulating the RCA 301.

"A micro-program simulation (i.e. emulation) is invariably more efficient than a macro-program simulation, since the simulation is being performed at the point of least difference between the computers."

(10)

BILLING, H. and HOPMANN, W., "Mikroprogramm-Steuerwerk", *Elektronische Rundschau* Nr. 10; 1955, pp. 349-359.

This paper, tutorial in nature, describes microprogramming as Wilkes defines it, and discusses read-only memory technology. In the original German.

(11)

BLANKENBAKER, J. V., "Logically micro-programmed computers", *IRE Transactions on Electronic Computers*, Vol. EC-9; June 1960, pp. 103-109.

"Simple computers exploiting the concepts of simulation and microprogramming are described. Logical rather than arithmetic microprogramming operations are employed for generality and greater simplicity. Design techniques are given for computers employing only multiple-bit time delays (i.e. no flip-flops or toggles are used)."

SICMICRO Newsletter, Vol. 1,
No. 2; August 1969, p. 10.

(12)

BORSCHEV, V. B.; VASILEVSKIY, P. L. and KHOMYAKOV, M. V., "A programming micro-language for Ural-4", *Kibernetika*, No. 6; 1966, pp. 47-49 (Russian).

Machine: Ural-4

"The usefulness of developing a programming language for Ural-4 is justified. The following conditions are formulated which must be satisfied by the programming micro-language for Ural-4 and its translator: a substantial reduction of the programmer's work (compared to manual programming); the quality of programs constructed by the translator is not poorer than that of programs written by hand; the training of programmers with a good knowledge of Ural-4 should not exceed several days; the language admits of readjustment, extension, and inclusion in the automatic programming system; translation from other programming languages to the proposed one is simpler than translation to machine language; the work of programming the translator for this language should be minimal. The following basic properties of the microprogramming language for Ural-4 are pointed out: the instruction system coincides with that of Ural-4; the instructions consist of the operation code and certain names -- alphanumeric words -- used for locating the numbers taking part in the operation; symbolic addressing is used; the program can be divided into rows (connected sequences of instructions, joined by a single condition for execution); block programming is applied. An exact description of the language, its syntax, and the translation rules are not given."

G. V. in *Cybernetics Abstracts*,
No. 9; 1967.

(13)

BOUTWELL, E. O., Jr., "The PB 440 computer", *Datamation*, Vol. 5, No. 2; February 1964, pp. 30-32.

Machine: PB 440

The PB 440 is described as a microprogrammed computer with two applications: (1) as a control and arithmetic element in real-time systems, and (2) as a high performance calculator for general scientific purposes. The author states that a micro-program is much faster than a conventional subroutine.

Access to the read-only memory in the PB 440 is one microsecond, while the cycle time of the main memory is five microseconds.

(14)

BOUTWELL, E. O., Jr. and HOSKINSON, E. A., "The logical organization of the PB 440 microprogrammable computer", *AFIPS Conference Proceedings*; FJCC 1963, pp. 201-213.

Machine: PB 440

One of the original motivations for designing computers with a microprogramming control feature was to enable the programmer to direct the computer at a lower level of control than was possible with conventional, wired-logic computers. The logical structure of the PB 440, which fulfills this lower-level control requirement, is the topic of this paper.

The paper describes the general characteristics of the computer and also the different types of micro-instructions available. Included in the micro-machine is an automatic instruction counter, the P register.

An early time-saving characteristic, that of being able to initiate a read operation before finishing a restore operation in the main memory, is also introduced with the PB 440.

(15)

BREUER, M. A., "Adaptive computers", *Information and Control*, Vol. 11, No. 4; October 1967, pp. 402-422.

This paper discusses several possibilities for graceful degradation in adaptive computers. Graceful degradation occurs when a computer can continue to function, at some loss of accuracy or speed when a portion of the hardware is malfunctioning.

The author suggests the use of microprogramming: an alternate microprogram, not affected by a particular set of component failures can perform the failing function - at the cost of memory space.

(16)

BRILEY, B. E., "Picoprogramming: A new approach to internal computer control", *AFIPS Conference Proceedings*; FJCC 1965, Part I, pp. 93-98.

Picoprogramming recognizes a "correspondence between the pulse-programming requirements of a control section and the capabilities of a memory element known as MYRA. A MYRA memory is a MYRiAperture ferrite disk which, when accessed, produces sequential trains of logic-level pulses upon 64 or more otherwise isolated wires."

The main advantage of picoprogramming is that no clock is needed because each MYRA disk, as it completes its switching, causes the next instruction to be obeyed.

(17)

BUCKINGHAM, B. R. S.; CARTER, W. C.; CRAWFORD, W. R. and NOWELL, G. A., "The controls automation system", *Sixth Annual Symposium on Switching Circuit Theory and Logical Design*; October 1965, pp. 279-288.

Machines: IBM 360 Systems, Models 40, 50, 65 and 67.

"This paper discusses the specifications, programming implementation, and use of a design automation system that can be used in the development of any computer having read-only storage (ROS) controls.

This system, called the Controls Automation System (CAS), designed concurrently with the IBM System/360 was used in the development of Models 40, 50, 65 and 67. The current version of CAS is used by IBM to aid in the design of all systems with ROS controls.

The newly developed elements of this system are: 1) a programming language for describing the data flow of a machine; 2) a program for translating a variety of microprogramming languages into a common one, and 3) a set of computer programs enabling the machines operation to be simulated on an IBM 7090/1401 computer."

(From the paper)

(18)

CALHOUN, R. C., "Diagnostics at the microprogramming level", *Modern Data*, Vol. 2, No. 5; May 1969, pp. 58-60.

Machine: RCA Spectra 70/35

Diagnostic procedures for the RCA Spectra 70/35 are implemented with a special diagnose instruction, which allows the execution of individual microinstructions ("elementary operations" or "functions") directly from main memory.

Two approaches to diagnostic procedures are discussed, both based on the use of the diagnose.

The first method involves testing at the instruction level until a problem occurs, then switching control over to the diagnose instruction. Major drawbacks to this approach are that: 1) sometimes the incorrect instruction cannot be isolated, and 2) the number of registers and the size of the instruction repertoire makes the number of possibilities to be tested prohibitively large.

The second method is a building block approach, in which testing is accomplished completely at the function level. For example, the transfer bus is tested after assuming the reliability of two registers. Then, if the transfer of information from one register to the other fails, the transfer bus should be at fault. The routine will continue if the transfer information is valid. The aim of the diagnostic routine is to isolate and test as few unknowns as possible at one time to better pinpoint a malfunction.

(19)

CAMPBELL, C. R. and NEILSON, D. A., "Micro-programming the Spectra 70/35", *Datamation*, Vol. 12, No. 9; September 1966, pp. 64-67.

Machine: RCA Spectra 70/35

A concise paper on the advantages of microprogramming in the Spectra 70/35. The 70/35 elementary operations, which include register-to-register and variable operations, are discussed with an example given of the microprogram for the SS format MOVE instruction. This paper emphasizes the practicality of the read-only memory design method for both user and manufacturer.

(20)

CHU, Y., "A higher-order language for describing microprogrammed computers", University of Maryland Computer Science Center, Technical Report 68-78; September 1968.

Mr. Chu defines a microprogram as a series of binary numbers linked together. He has developed an ALGOL-type symbolic language to describe microprograms. This paper introduces the language by describing as an example a serial complement sequence using both normal computer techniques and microprogrammed control.

The difference between sequential logic control and microprogrammed control is (in this particular system) a trade-off between time and flexibility. Normal control in the example requires four elementary operations, microprogramming requires six. However, the latter allows flexibility because each bit in the instruction word can cause one micro-operation to be executed. Because of this opportunity for parallel operations, only three instruction words are actually needed.

Two drawbacks appear in this specific microprogram. Considerable time can be lost in addressing the control memory; and the multiplicity of micro-operations would require very long control words. Trade-offs are possible, but at the cost of the full flexibility of the bit-per-operation concept.

A simulator has been constructed to test this higher-order language and a print-out is provided.

(21)

CONROY, E. D., "Microprogramming", *Preprints of papers presented at 16th National Meeting of the ACM*; Los Angeles, September 5-8, 1961.

Machine: IBM 7950 System

E. D. Conroy defines microprogramming as the "direct control by the programmer of more than one echelon in the hardware hierarchy." Microprogramming allows the programmer direct control over the elementary operations.

Only the abstract and summary of the paper is presented.

(22)

CONROY, E. D. and MEADE, R. M., "A microinstruction system", *Preprints of papers presented at the 16th National Meeting of the ACM*; Los Angeles, September 5-8, 1961.

This paper describes a computer organization in which only the more complex operations (for example, variable length add) exist as instructions. Each such (primary) instruction is followed by some microinstruction words (up to 256) which specify microinstructions (for example, load accumulator with zero) and the conditions which must exist in order for these instructions to be executed. If such a condition arises, the execution of the primary instruction is interrupted to permit execution of the microinstruction depending on the conditions.

In this brief summary, discussion of the technique is omitted.

(23)

CONTI, C. J.; GIBSON, D. H. and PITKOWSKY, S. H., "Structural aspects of the System/360, Model 85", *IBM Systems Journal*, Vol. 7, No. 1; 1968, pp. 2-21.

Machine: IBM 360/85

This paper has as its purpose the introduction of the IBM 360/85, both in design and performance, and assumes knowledge of the IBM 360 system.

The first section deals with the design of the computer, and introduces a high-speed buffer called a cache. The cache provides for more storage space in the main memory by holding those portions of main storage that are currently in use.

(24)

DECISION SYSTEMS, INC., "Interim status report No. 1", TR 517-1; August 11, 1968.

The ideal computer, according to this article, is one that must be able, via microprogramming, to encompass a variety of instruction sets and provide an easy means of switching these sets. Such a system could have the capability of general emulation.

To attain this goal, this interim report studies the standardization of microprogramming techniques and attempts to incorporate these benefits: 1) ease of microprogram preparation and the ability to load a control memory with different micro-instructions in the field, 2) designing compilers and check-out systems for the microprogram, 3) create better debugging devices for microprograms, and 4) create an "ideal" logic simulator which analyzes instruction sets before they are committed to hardware.

The theoretical microprocessor would have three major units, each able to function independently. They are an internal processor, a main memory controller, and an I/O controller. Each of these units would be controlled by the read-only control memory. A description of the read-only memory and the planned microinstruction format is also given.

(25)

DREYER, L., "Principles of a two-level memory computer", *Computers and Automation*, Vol. 17, No. 5; May 1968, pp. 40-42.

Machine: Elbit 100

This paper describes the Elbit 100 computer. It argues the merits of a two-level memory computer, one level of which is the normal storage unit, while the other is a read-only memory. A speed increase due to the read-only memory occurs not from faster computing circuits, but from operational instructions built into the hardware for more efficient sequences of orders.

A good schematic diagram of a read-only memory control is included in the article. One memory is a standard random-access core store with a two microsecond cycle time while the other is a micro-programmable read-only store with 400 nanoseconds access time. The Elbit 100 has perhaps the simplest micro-machine of any existing computer.

(26)

EMELYANOV-YAROSLAVSKY, L. B. and TIMOFEEV, A. A., "Microprogram control for digital computers", *Proceedings IFIP Congress 62*; pp. 567-569.

This brief but excellent article provides a description of micro-instructions (the author terms these micro-orders), the information that each micro-instruction must contain, and some of the applications of microprogram control.

The three types of information required are: (1) operational, which controls the execution of a statement; (2) address information, which in some manner contains the address of the next micro-instruction; and (3) time information, which determines the processing time of the micro-instruction.

Under the heading of applications, the author's main point is that microprogram control can execute complex operations more efficiently by enabling the programmer to take into account the special properties of the algorithm, as well as using simpler circuits to generate the control circuit.

Finally, the author hypothesizes that two or more microprogram control computers can work together efficiently.

(27)

FLYNN, M. J. and MACLAREN, M. D., "Microprogramming revisited", *ACM Conference Proceedings*; 1967, pp. 457-464. (Essentially identical to the Argonne National Laboratory internal document Technical Memorandum No. 134, dated 1 June 1967, same title and authors.)

As in many of the better technical papers, the authors early and concisely state their aims:

It is the objective of this paper to briefly trace the history of the idea (microprogramming) and the difficulties involved with defining or implementing it. In doing this, we first consider the general control problem and instruction formats. Next, storage implementations of the control function are considered and a restricted definition of microprogramming is proposed. This is then evaluated from a technological, architectural and programming point of view. We hope to show that our (demanding) definition of microprogramming is now technologically feasible and attractive from systems consideration.

In addressing these objectives the authors have produced a worthwhile paper.

The development supporting the definition of microprogramming is clear and to the point. Consideration of technological, architectural, and programming implications of this early development is less successful, but still worthy of attention. On occasion the authors resort to designing when they should be stating requirements or categorizing potential properties, but their design sketches hit a nice level of complexity for their presentation purposes. Fired with enthusiasm, Messrs. Flynn and Maclaren are guilty of identifying and discussing only some of the problems and potentials of microprogramming (according to their definition, which assumes a read/write control store).

The problems of microprogramming on the level contemplated by the authors are those of programming. No observation is made that the control store becomes yet another level to be (dynamically) managed within the hierarchy; rather, specific ad hoc management techniques are presented. Similarly, deadlock potential in a multi-programmable microprogram is ignored, and protection problems are considered by design sketch rather than by requirement specification.

In perspective, however, the paper is of interest because of its structured, considered approach to some real problems, including some credible comments on the software aspects of microprogramming -- a subject too often completely ignored in the literature.

The references are adequate and supportative, and the general form of presentation satisfactory (in spite of several typographical errors).

(28)

GERACE, G. B., "Microprogrammed control for computing systems", *IEEE Transactions on Electronic Computers*, Vol. EC-12; December 1963, pp. 733-747.

Starting with Wilkes' logical design of a control unit, this paper develops some variations and extensions. Specifically, it deals with problems concerning the conditions intervening at the end of a micro-order to determine the next one, with timing of micro-operations and also time-sharing and multiprogramming.

(29)

GLANTZ, H. T., "A note on microprogramming", *Journal of the ACM*, Vol. 3, No. 2; April 1956, pp. 77-84.

This article, written in August 1955, is an attempt to provide a rough outline of some conditions under which microprogramming might be a useful technique. Two methods of constructing a micro-programmed machine, a plugboard control and an internal relay setting control, are discussed in Section IV of the paper.

Efficiency is touted as the basis for a decision in favor of using micro-operations. The problems encountered in microprogramming are discussed in general from both an engineer's and a programmer's viewpoint.

(30)

GRASSELLI, A., "The design of program-modifiable micro-programmed control units", *IRE Transactions on Electronic Computers*, Vol. EC-11; June 1962, pp. 334-339.

In this paper a design is given for a program-modifiable control unit which does not require the usual high speed control store. A wired-in control memory contains microinstructions stored without repetitions. These are executed by means of words stored in core memory, each word containing the addresses of several microinstructions plus looping information. The logic necessary to sequence these words is quite complicated. Furthermore, the user is restricted to the wired-in microinstructions.

"The author refers to the reviewer's expressed opinion that a program-modifiable microprogram system would probably not justify itself in practice, but suggests that in the area of non-numerical information processing of the 'production' type this may not be so. The case is not, however, argued in detail, and the author soon passes to a description of a proposed method of designing a program-modifiable control unit.

In the conventional microprogramming system a read-only memory is used to hold the program of micro-orders. In the proposed scheme the read-only memory contains the micro-orders arranged in a random order, each order used in the microprogram appearing once only. There is a secondary memory of normal erasable type containing words into which information about the microprogram is packed. These words are processed in order by the sequencing circuits, and each causes a sequence of micro-orders to be extracted from the read-only memory and executed.

The scheme is ingenious but complicated. The author realizes this, and he also draws attention to the other major disadvantage, namely, that the microprogrammer is restricted in his choice of micro-orders to those which are wired into the read-only memory. This might seriously prejudice the main object of the scheme, which is to get closer to ultimate efficiency by being able to choose an order code optimum for the application in hand."

M. V. Wilkes, Cambridge, England,
Computing Reviews; 1963, #4166,
p. 129.

(31)

GREEN, J., "Microprogramming, emulators and programming languages", *Communications of the ACM*, Vol. 9, No. 3; March 1966, pp. 230-232.

"This paper is concerned with the no-man's land between, or common to, machines and languages; the land of soft hardware and hard software. We know the problem of finding a machine in terms of which programming languages can be defined; what of the problem of finding a language in terms of which machines can be defined? There is a need for a metalanguage in which no distinction need be made between hardware and software, between machine and language, so that the gap which exists in practice can be bridged in discussion.

Some confusion to the reader is likely on first reading owing to the fact that the word 'automaton' is incorrectly printed as 'automation' in a number of places."

F. G. Duncan, London, England,
Computing Reviews; September-
October 1966, #10,450, p. 409.

(32)

GREENBERG, M. and WEGBREIT, E., " μ -212: A microprogrammed computer", Unpublished class notes, Eng. 212, Harvard University; May 1968.

Machine: μ -212

This set of class notes for "Engineering 212" presents the detailed design of a microprogrammed computer, the μ -212, which was created to highlight microprogramming aspects. Descriptions of the central processing unit, basic instruction set, and control unit (read-only) are listed. Three appendices are also provided:

1. Hardware drawings
2. Various microinstructions
3. Operation codes

The read-only memory has a 170 nanoseconds cycle time divided into three phases. Period one executes the current microinstruction and loads the control memory address register with the next microinstruction. Period two starts the reading of the new instruction from the address register, and period three places the instruction into the memory buffer register.

The authors claim the novelty of their read-only memory control design, but do not elaborate.

(33)

HAGIWARA, H.; AMO, K.; MATSUSHITA, S. and YAMAUCHI, H., "The KT Pilot computer - A micro-programmed computer with a phototransistor fixed memory", *Proceedings IFIP Congress 62*; pp. 684-687.

Machine: KT Pilot

This paper introduces a computer designed jointly by Kyoto University and Tokyo Shibaura Electric Company Ltd. The system design, circuit elements and physical construction of the computer are briefly described. The order code of the machine is under the control of the programmer.

The fixed memory is unique in that it has been constructed with phototransistors. A punched card template is placed over an array of phototransistors and, by means of projected light, a path in a matrix is either opened or remains closed.

The output of the read-only memory is used to open and close gates (1 gate per bit organization). There are 256 eighty bit words in the ROM.

(34)

HARRAND, Y., "Evolution of microprogramming concepts", *Proceedings 3rd AFCALTI Congress of Computing and Information Processing*; Toulouse, 1963, pp. 187-190.

"The author defines a microprogrammed machine as one that has only those orders wired in that can be done in one machine cycle. Actual executed instructions are built up out of these basic operations. Generally, each 'macro-instruction' is executed as some sort of a hardware subroutine. The paper discusses the advantages of such machines, especially in the case where the user is allowed to design his own hardware subroutine. Of course, this technique is used in the hardware emulators of other machines used in IBM's System/360 machines."

J. E. Denes, Upton, New York,
Computing Reviews; March-April 1966,
#9474, p. 189.

(35)

HAWRYSZKIEWYCZ, I. T., "Microprogrammed control in problem-oriented languages", *IEEE Transactions on Electronic Computers*, Vol. EC-16, No. 5; October 1967, pp. 652-658.

Machine: CIRRUS

The application of microprogramming to problem-oriented languages is described in terms of a simulated analog system on a digital computer. In the system described a problem defined by differential equations is drawn up in the form of an analog diagram. The system allows the machine assembly code to be freely mixed with the analog-oriented source language input in coding the supervisory and interrupt facilities.

(36)

HECKELMAN, R. W., "Self-checking redundant microprograms", *IEEE Computer Group Repository R67-31*; August 1966.

"Quick, comprehensive error detection is a requisite of high reliability."

Hardware solutions are either too expensive or incomplete, while software solutions are both incomplete and result in severely slowed performance.

"If redundant software instructions are implemented as microinstructions, then the speed of checking is greatly increased."

"In this report, redundant algorithms and corresponding microprograms are presented and evaluated for detection of both transient and steady errors in parallel arithmetic and logical operations. The results are quite promising with respect to potential speed of performance, percentage of added hardware, and comprehensiveness of error detection."

(37)

HELLERMAN, L. and HOERNES, G. E., "Control storage use in implementing an associative processor for a time-shared processor", *IEEE Transactions on Computers*, Vol. C-17, No. 12; December 1968, pp. 1144-1151.

Machine: IBM 360/40 (CP-40)

This paper discusses the use of the Associative Memory (AM) used to implement the Cambridge System, a modified 360/40. The Cambridge System allows up to 15 users to time share what appears to each user to be a standard, complete, but slow 360/40. The AM is used for automatic address translation, supporting a paging concept; it is controlled via microprogram.

The AM is functionally described, as are the controlling microprograms.

(38)

HILL, R. H., "Stored logic programming and applications", *Datamation*, Vol. 10, No. 2; February 1964, pp. 36-39.

The author evaluates "...the contributions of stored logic to computer design...from the viewpoint of programming and applications."

He concludes that the techniques have not proved themselves, that "...the developers...found they had put together general purpose devices directly competitive with more conventionally organized computers in any given applications area."

"...Stored logic loses its significance when a stored logic computer is called upon to compile and execute FORTRAN, COBOL, NELIAC, JOVIAL, etc."

"...the future of stored logic as a design technique is somewhat cloudy."

(39)

IBM, "Data processing: Microprogramming PI course", IBM #221-0083.

Machine: IBM System 360

This course is designed to introduce the student to an in-depth study of microprogramming for the IBM System 360 series. It describes the engineering techniques employed in microprogramming, but the primary concern is for the programmer's point of view.

Section I includes an introduction to read-only storage and control points. Section II contains ideas on the microprogram and control file memories. Section III is concerned with Controlled Automated System (CAS) logic diagrams.

(40)

INGALLS, R. A., "Logical design of a microprogrammed special-purpose computer", United States Naval Postgraduate School; December 1966.

Machine: UNIVAC 1830 Avionics Computer

This thesis is a comparison between a conventional computer, the UNIVAC 1830, and an experimental computer similar to the UNIVAC 1830 which uses a microprogrammed control unit.

The first section contains a general outline on the difference between conventional and microprogram control, followed by the test itself. The test results clearly show that the microprogrammed computer is faster; for example, the add operation under microprogramming takes 275 microseconds, while conventional means would take 1016 microseconds. This test, operated under hypothetical conditions, appears to contribute to the theory that microprogramming can contribute to a more efficient computer.

(41)

IVERSON, K. E., "Microprogramming", *A Programming Language*, Chapter 2, John Wiley and Sons, Inc.; New York, 1962, pp. 71-104.

Machine: IBM 7090

This chapter from Iverson's book defines microprogramming to include paper descriptions of a computer's machine language which could be used to bridge the gap between the computer design and the logical circuit designer. Iverson proceeds to describe the 7090 machine language in terms of more elementary operations.

(42)

KAMPE, T. W., "The design of a general-purpose microprogram-controlled computer with elementary structure", *IRE Transactions on Electronic Computers*, Vol. EC-9; June 1960, pp. 208-213.

Machine: SD-2

This paper discusses the logical design of a binary, parallel, real-time computer. A description of the type of machine wanted is followed by the design decisions generated by the description. The contents of the micro-controlled logic generation are discussed, with reasons given for their incorporation into the scheme.

(43)

KLEIN, S. and SCHWARTZ, S., "Model 4200-8200 read-only memory control logic", *Honeywell Computer Journal*; Winter-Spring 1968, pp. 25-33.

"This paper summarizes the principal characteristics and the advantages of the read-only memory (ROM) as a control element in the 4200/8200 processors. The functions of both the address-generator ROM and the arithmetic unit ROM in the Model 8200 are described. Bit steering is discussed as a decoding scheme used to reduce the size of the ROM in the Model 4200." (Author)

Described with facility poor even for a house journal, bit steering is an encoding scheme intended to provide "a method by which one could change the way ROM bits were interpreted without having to rewire the machine." The full flexibility achieved was not what might have been desired. Consider the number of μ -operations which may be explicitly and concurrently mentioned "by (encoded) name" by b bits. On one hand we may associate one bit for each of b μ -operations, allowing concurrent mention of b distinct μ -operations. Alternatively, one might associate a μ -operation with each of the 2^b states uniquely expressable by the b bits, allowing concurrent mention of only one of the 2^b possible μ -operations.

"Bit steering" chooses a middle ground. Given a field of $n + b$ bits, let the n bits select ("steer to") one of 2^n sets of association maps for the b bits. Each association map pairs each of the b bits to one μ -operation. One has the potential of addressing up to $2^n \times b$ μ -operations, b at a time. In practice, the association maps are not mutually exclusive.

(44)

LAWSON, H. W., Jr., "Programming-language-oriented instruction streams", *IEEE Transactions on Computers*, Vol. C-17, No. 5; May 1968, pp. 476-485.

In a fourth generation computer with a changeable control store, microprograms can be used to implement a variety of instruction sets. It then becomes practical to translate a higher level language into some intermediate language which efficiently and conveniently expresses the higher level language, and then interpret the intermediate language by means of a microprogram. Six candidates for this intermediate language are defined. These candidates are then compared and evaluated with respect to conciseness, complexity, dynamic capability and flexibility.

(45)

LEVY, L. S., "State of the Art of Microprogramming", Report No. TOR-469(5710-01)-1, Aerospace Corporation, El Segundo, California; 15 April 1965.

This report covers the history, concept and advantages of microprogramming until 1965. The history is somewhat sketchy and only the fact that Wilkes invented the term is discussed in detail. The concept, again that of Wilkes, is shown to be a dual-matrix control which handles elementary operations.

Microprogramming is described as "a technique for systematizing the design of the central sequencing and control in a digital computer". The incorporation of microprograms in groups of computers is then given, including the Spectra 70 systems and the IBM 360 systems.

The last section of the paper deals with the design features of microprogramming, such as a modifiable order code, a late design freeze, and the improvement of speed. Some reference is given to the microprogram's role in multiprogramming, but this is very sketchy.

(46)

LIVERANCE, H. H., "A System/360 profile -- Read-only storage", IBM Technical Information Exchange #Z77-5170; June 1965.

Machine: IBM 360 Systems

This paper presents a factual, concise and in-depth explanation of the concept of a read-only memory, with particular reference to the read-only memories used in the IBM 360 systems.

The two technologies used at IBM for the read-only storage are: (1) transformer read-only storage (TROS), and (2) capacitor read-only storage (CROS). The fact that these are permanent, but easily replaceable, devices is their advantage over conventional computer devices.

The functions of the ROS program are explained, along with their relationship to the macro-instructions they are executing. The distinction between "interrupt" and "trap" is made explicit. "A Trap is a hardware-initiated asynchronous transfer to some other point in the Microprogram. An Interrupt is a ROS-initiated, asynchronous transfer to some other point in the Macroprogram."

(47)

MATSUSHITA, S., "A microprogrammed communication control unit, the TOSBAC DN-231", *Preprints of papers presented at IFIP Congress 68, Edinburgh; Booklet D: Hardware I*, pp. D108-D112.

Machine: TOSBAC 3400

The TOSBAC DN-231 is a microprogrammed communication control unit designed to provide more efficient passage of I/O information to and from a general computer system. It contains a multiplexor channel controlled by microprograms.

This article provides basic reasons for using microprogramming: 1) decreased software overhead, 2) the flexibility needed to incorporate new control methods and new message formats without changing hardware, 3) easier maintenance, and 4) the ability to use special functions to save trivial software tasks.

A number of special functions are realized by microprograms. One such function provides for automatic data-chaining, concatenating buffers automatically to accept data streams. Another instruction, called POL 0, polls all the lines in a designated range. POL 0 polls only those lines used more frequently. By combining these two instructions, using POL 0 only occasionally, the user can decrease overall polling time.

This article, without going into great detail, points out some of the potential benefits of microprogramming a communication channel.

(48)

McCLURE, R. M. and DAVIS, R. D., "A small microprogrammed computer", IEEE Computer Group Repository R-69-58; 1969.

The paper is primarily a description of the SCC 4700, "...a 16-bit general purpose digital computer that features a microprogrammed control section. The paper gives the details of the microcode sequences. The paper also gives a few of the authors' opinions of the microprogramming technique for computer control sections."

The authors conclude that "microprogramming is not a panacea...microprogramming must be considered as simply another way by which computers can be implemented."

(49)

McCORMACK, M. A.; SCHANSMAN, T. T. and WOMACK, K. K., "1401 compatibility feature on the IBM System/360 Model 30", *Communications of the ACM*, Vol. 8, No. 12; December 1965, pp. 773-776.

Machines: IBM 360/30 and 1401

This paper describes the hardware available for the IBM 360/30 which makes the emulation of an IBM 1401 possible. Special consideration is given to the read-only memory which contains the microprogram to do the emulation. The flexibility allowed through microprogramming in reference to the instruction counter, sense switches and stop routine is discussed, and the author concludes that the inherent flexibility of the read-only storage allows the emulation of an IBM 1401.

(50)

McGEE, W. C., "The TRW-133 computer", *Datamation*, Vol. 5, No. 2; February 1964, pp. 27-29.

Machine: The TRW-133 (BR-133)

"This is a very clear and concise description of a small general-purpose machine that was built with a fairly large set of primitive orders that can be combined into subroutines by the user, and are executed under the control of a sequence of interpreted macro-orders.

The author, in addition to his clear expression, is to be congratulated on his restraint in avoiding the term 'micro-programming'.

The machine was designed for shipboard NTDS usage, at far greater speed, reliability and I/O processing than that of its predecessors designed by the same company. With the exception of the interpretive execution of micro-orders, its memory, logic, and registers are conventional."

Herbert M. Teager, Cambridge, Mass.,
Computing Reviews, Vol. 6, No. 2;
March-April 1965, #7440.

(51)

McGEE, W. C. and PETERSEN, H. E., "Microprogram control for the experimental sciences", *AFIPS Conference Proceedings*; FJCC 1965, pp. 77-91.

Use of microprogrammed control units is suggested for interfacing experimental apparatus or computer peripheral devices with a computer. "...conventional design process is straight forward, it has the inherent disadvantage that it must be repeated for each new configuration...a single design schema which was sufficiently general to accommodate a wide variety of computers and external equipment, and which could be quickly and easily particularized to meet the requirements for specific controllers." The internal design of such a controller is described in great depth.

The IBM 2841 disk, drum, data cell controller is cited as one example, and a microprogram for controlling PEPR, "...a computer-controlled CRT scanner used to automatically measure bubble chamber tracks which have been recorded on film", as another.

(52)

McGEE, W. C. and PETERSEN, H. E., "Microprogramming for data acquisition and control", *IEEE Transactions on Nuclear Science*, Vol. NS-12, No. 4; August 1965, pp. 310-320.

Essentially equivalent to McGEE, W. C. and PETERSEN, H. E., "Microprogram control for the experimental sciences", *AFIPS Conference Proceedings*; FJCC 1965, pp. 77-91.

(53)

MEGGITT, J. E., "A character computer for high-level language interpretation", *IBM Systems Journal*, Vol. 3, No. 1; 1964, pp. 68-78.

This paper discusses the design of an experimental character-processing computer for the interpretive execution of higher-level language programs.

The design specifies a 100 nanosecond instruction cycle for the microprogram instructions stored in a read-only memory, a fast memory for intermediate "scratch pad" computation, and input/output through a conventional computer coupled to a two microsecond main memory.

(54)

MELBOURNE, A. J. and PUGMIRE, J. M., "A small computer for the direct processing of FORTRAN statements", *Computer Journal*, Vol. 8; 1965, pp. 24-27.

On this small, special-purpose computer, FORTRAN programs are compiled and executed. The big difference lies in the micro-programming of both operations.

In the first phase of compilation, statements are entered at an on-line typewriter. Key words (DO, DIMENSION, etc.) require only one key to be depressed - the microprogram echoes the full word. Statements are collected, error-checked and stored in digested form. The second phase completes the compilation process. Expressions are transformed into Reverse Polish form and space is allocated.

Program execution is controlled in the microprogram using an Instruction Stack and a Data Stack. The Instruction Stack provides program control, particularly useful in controlling nested DO statements. The Data Stack is used by the microprograms corresponding to each statement, particularly for function and expression evaluation.

A DEBUG switch allows on-line tracing of the execution of each statement, by the logging of its statement number.

Using a standard FORTRAN compiler on another machine, and hand coding the same machine, the authors conclude that their FORTRAN lies somewhere between the other two techniques in performance.

(55)

MERCER, R. J., "Micro-programming", *Journal of the ACM*, Vol. 4; April 1957, pp. 157-171.

The first section of this article presents a general outlook on the concept of microprogramming, incorporating the work of M. V. Wilkes and H. T. Glantz.

Microprogramming is defined as "...the technique of designing control circuits of an electronic digital computer to formally interpret and execute a given set of machine instructions, Mi, as an equivalent set of sequences of micro-operations, elementary operations that can be executed in one pulse time."

The author gives an example of an adder in a hypothetical microprogram, explains how it works and shows why it would take less time than an ordinary add instruction. The other sections of the paper deal with rather unrelated specific problems.

(56)

MILLER, E. H., "Reliability aspects of the Variable Instruction Computer", *IEEE Transactions on Electronic Computers*, Vol. EC-16, No. 5; October 1967, pp. 596-602.

This article stresses the use of the variable instruction concept to enhance reliability by means of controlled graceful degradation. When a component fails, the set of microprograms which implement the macro-operations dependent on that component are replaced with a new set not so dependent. Processing can then continue, perhaps less efficiently.

Much of the article is descriptive of the logic used to arrive at the VIC design; embedded in this is most of the design itself.

See also Spence, A. L., "Hardware and Software Interactions; A Machine Organization Solution, The Variable Instruction Computer".

(57)

MORRISON, G. E., "Investigation of micro-programming as a technique in the control of digital computing systems", (Thesis), University of California at Berkeley; 1958.

"The micro-program technique produces desired external operations in a digital computing system by sequences of internal operations controlled by a stored program of internal orders called 'micro-orders'. The various 'micro-program' sequences are selected by a second level of command, called 'sequence command'. This evaluation of the technique consists of an investigation of the effect of use on hardware complexity, computing speed, and ease of use.

Investigation of design considerations indicated the choice of a system consisting of an arithmetic section designed to be controllable by micro-orders, a memory which stores the micro-order sequences, a second memory which stores the sequence commands, a third memory which stores the problem data, a unique modified index register system for memory address control, and appropriate buffer and control registers.

It is shown that the arithmetic section could be achieved with no more functional elements than the minimum required in any system for the selected arithmetic operations and computing times. No unusual requirements were found for the problem data and sequence memories other than the need for independent control. Investigation of the micro-program memory indicated that a capacity of 10^6 bits with a speed comparable to the data memory would be desirable.

By investigation of a 'standard' operation and matrix multiplication program, it is found that the micro-program system was significantly faster than a 'typical' conventional system. However, it was about as fast as a conventional system modified to provide more effective subroutine operation with a resulting complexity which was comparable to that of the micro-program system..."
(From the abstract)

Computing Reviews, Vol. 9, No. 1;
January 1968, #13,456, pp. 40-41.

No legible copy of this thesis was available for review.

(58)

OPLER, A., "Fourth-generation software: The realignment", *Datamation*, Vol. 13, No. 1; January 1967, pp. 22-24.

This paper deals with the future of computers. Mr. Opler projects that microprograms will be used more frequently. He predicts that fourth-generation hardware will use large-scale integration (LSI) logic specialized by microprogramming. Mr. Opler hypothesizes the speed of three elements in a fourth-generation computer: circuitry (LSI) takes 5 nanoseconds, the micromemory access takes 10 nanoseconds, and main memory access 400 nanoseconds. Therefore, microprogram interpretation of instructions is feasible. He also introduces the term "firmware" to designate microprograms in the control memory. He expects the most important potential of the microprogram to occur with the advent of a slow-write/fast-read memory. This would enable the programmer, via a special set of punch cards, for example, to change the contents of the micromemory, thus permitting him to define his own instruction sets.

(59)

OPLER, A., "New directions in software 1960-1966", *Proceedings of the IEEE*, Vol. 54, No. 12; December 1966, pp. 1757-1763.

Among other topics, the effects of microprogramming and its potential are examined. Designers have begun to study it in depth because several micro-instructions of 10-100 nanoseconds each can be executed in less time than access to the main memory (500-2000 nanoseconds).

The concept of virtual memory with peripheral devices containing more storage is discussed in relation to microprogramming by showing how micro-operations can decrease access time. This is a short but excellent summary of the advantages of microprogramming.

(60)

PATZER, W. J. and VANDLING, G. C., "Systems implications of micro-programming", *Computer Design*, Vol. 6, No. 12; December 1967, pp. 62-66.

This article gives a description of the functions of micro-programmed control and makes a comparison between general-purpose and microprogrammed machines. It dwells on microprogramming from both the engineer's and programmer's viewpoint; and, without going into any depth, does a creditable job. It says that a microprogram may consist of: (1) a fixed sequence of words containing conditional micro-orders; (2) a variable sequence of words containing unconditional micro-orders; or (3) a combination of both.

A hypothetical comparison is drawn in performing a square root with conventional and microprogrammed instructions.

(61)

RAKOCZI, L. L., "The computer-within-a-computer: A fourth generation concept", *Computer Group News*; March 1969, pp. 14-20.

Machines: Standard Computer Corporation IC 3000, 4000, 7000

"The fourth generation computer is a systems-architecture-oriented device which stresses changes in structure, organization and utilization of computer techniques and programming skills." Mr. Rakoczi thus defines his concept of the fourth generation computer. Since fifty per cent of the article is a sales brochure, the convenience of having the only fourth generation computer is obvious.

The division between third and fourth generation computers seems to lie in the use of microprogramming. The microprograms resident in third generation computers are a replacement for controlled logic. They are devices to simplify and organize design procedures and aid in maintenance problems. The microprogram in a fourth generation computer system is in itself an inner computer, designed not only to help the engineer, but also the programmer. The inner computer controls all communication between the functional stations of the system, aiding in efficient subroutine handling.

Perhaps the most sweeping claim made is that the inner computer, along with a process called MINIFLOW, allows for complete emulation. Even the author admits that only one type of computer can be emulated at any given time. No examples are given of the time factor involved if more than one machine is to be emulated during the same process; and no discussion is given as to the effect when the emulated machine has a different length word from the micro-machine.

(62)

REBOULOT, M., "Extracteur microprogramme d'information radar", *L'Onde Electrique*, Vol. 48, No. 491; February 1968, pp. 138-140.

Machine: IBM 360/30

EMIR (acronym from the title) is a real-time microprogrammed collector of radar data, apparently implemented using an IBM 360/30 CPU. EMIR also includes communication circuits with the radar, and another, possibly remote, more powerful computer, which might perform functions such as trajectory plotting.

The author defines microprogramming by example of an add operation, and does so classically: each microinstruction, operating in a single cycle time, opens or closes gates which correspond to the bits in the microinstruction.

The 360/30 condenser memory is briefly described, as are the IBM programs for developing microprograms.

The author is most taken with the flexibility which would allow rapid adaptation to new developments in radar technology or usage.

The article appears in the original French.

(63)

ROSE, G. A., "'Intergraphic', A microprogrammed graphical-interface computer", *IEEE Transactions on Electronic Computers*, Vol. EC-16, No. 6; December 1967, pp. 773-784.

Machine: Intergraphic

This paper describes a proposed microprogrammed interface computer to link many (from 13 to 50) general-purpose graphical terminals to a central processor. The digital computer is fast (3-5 nanoseconds for integrated circuitry and 100 nanoseconds access time to the read-only memory) and versatile because of its read-only memory.

The control section comprises a ROM, addressed from a "read" register, a micro-routine link register Q, decoding and timing logic, and a bit-wise programmable register F. One of the microprograms is a small Cartesian vector program with 28 micro-orders.

(64)

ROSEN, S., "Hardware design reflecting software requirements", *AFIPS Conference Proceedings*; FJCC 1968, pp. 1443-1449.

The author introduces microprogramming as one of a number of hardware developments (the first one he lists) which has "contributed to the development and the performance of multiprogramming and multi-processor operating systems and compiling and translating systems."

A conventional definition is given, and the relation to simulators/emulators is pointed out.

(65)

SCHLAEPPI, H. P., "Microprogrammed computer control", *IBM Technical Disclosure Bulletin*, Vol. 5, No. 9; February 1963, pp. 45-46.

Machine: IBM 360 Series

This short (two page) article describes the use of a special return register along with a counting address register that eliminates the need for an explicit micro-address for each new microinstruction. A description of the execution of a micro-operation in each of four modes follows. The modes are: normal, prepare-return, execute-return and jump. The only mode which requires an explicit micro-address is the jump mode, in which a field of the microinstruction is interpreted as the next micro-address.

(66)

SEMARNE, H. M. and McGEE, W. C., "Stored logic computing", *Preprints of papers presented at the 16th National Meeting of the ACM*; September 5-8, 1961.

A stored logic type of computer is one in which the basic logic expressed by micro-commands is implemented by hardware, and all computer instructions above this building block level are programmed. The authors discuss the merits of the stored logic computer in relation to ease of learning, ease of modification, speed, and versatility, but do not go into much depth.

(67)

SEMARNE, H. M. and PORTER, R. E., "A stored logic computer", *Datamation*, Vol. 7, No. 5; May 1961, pp. 33-36.

Interpretation as a technique leads to performance losses of a factor of 10 to 15. The stored logic concept, say the authors, offers "interpretation without deceleration".

"One may consider computer instruction types as rungs on an 'abstraction ladder' rising from machine language toward natural human language, with distance from the bottom denoting level of abstraction with respect to the computer logic.

The bottom rung of the instruction ladder is held by the primary building-blocks of all types of computer instruction, namely micro-commands which cause the machine to open gates, trigger flip-flops, and perform other useful circuit functions.

In the Stored Logic concept, much of the wired logic which mechanizes the machine-language instructions customarily found in an instruction list is disregarded, and only the basic logic expressed by the micro-commands is implemented by hardware. The logic for any instructions above the micro-command level is stored in the computer memory in the form of a micro-program, i.e., a sequence of micro-commands. Hence, in the Stored Logic concept, any type of instruction can be made up out of micro-commands and stored in memory access as a macro-instruction. Thus, conventional machine language differs from an interpretive system only by a degree of abstraction..."

(68)

SLOBODYANYUK, T. F., "A Method of Minimizing Microprograms", AFSC Foreign Technology Division, FTD-HT-23-600-67, AD 662 821; 27 July 1967.

This article gives an algorithm for reducing a sequence of elementary micro-operations into a sequence of microinstructions. A microinstruction contains one or more elementary micro-operations, all of which can be performed in parallel. The desired microinstruction sequence attains a maximum speed coupled with a minimal amount of apparatus. The algorithm produces this sequence by doing a backward analysis of the contents of the registers. Experiments showed that in the new sequence the execution time is shortened by 6-12%; its size is 23-29% smaller than the original sequence.

(69)

SPENCE, A. L., "Hardware and Software Interactions: A Machine Organization Solution, The Variable Instruction Computer", Radio Corporation of America; 23 September 1966.

Machine: RCA VIC

This paper, written during the embryonic stages of the VIC computer development, contains the elementary concepts of the microprogramming involved in it. The high speed memory, with a "scratch-pad" section, consists of two modules, each containing 256 x 38 bit words. On Page 5 is an excellent example of how a macroinstruction (add) is carried out by a sequence of micro-orders. Many reasons for the use of microprogramming are given including flexibility, utility and speed.

See also Miller, E. H., "Reliability Aspects of the Variable Instruction Computer".

(70)

STEVENS, W. Y., "The structure of System/360; part II - system implementations", *IBM Systems Journal*, Vol. 3, No. 2; 1964, pp. 136-143.

Machines: IBM 360/30, 40, 50, 60, 62, 70

This paper discusses the system implementations of the 360 series and measures performances in four areas: 1) main storage, 2) central processing unit, registers and data paths, 3) sequence control, and 4) input-output channels. Each of these is then further categorized into three factors: basic speed, size, and degree of simultaneity.

The read-only storage control is discussed under the major heading of sequence control. Besides listing certain characteristics of the read-only storage in the individual models in the 360 series, the author states, "This microprogram control...is the only method known by which an extensive instruction set may be economically realized in a small system." The IBM 360/60, 62, is presented as proof of this. Originally designed with conventional control logic, the IBM 360/60, 62, was altered for microprogramming control when its instruction set required too many circuit modules.

(71)

STRINGER, J., "Microprogramming and the choice of order code", *Proceedings Symposium on Automatic Digital Computation*; London, 1953, pp. 71-74.

Written in the first stages of the evolution of microprogramming, this paper first overviews the original view of the concept, then questions the choices of order code that could be built into the microprogram.

Most of the article expounds certain advantages that seemed to provide a raison d'etre for microprogramming. These include late design freeze, easier and faster floating-point arithmetic, and more efficient control methods.

This paper is interesting from the standpoint of examining the early endeavours into the field of microprogramming; but it does not introduce the reader into any in-depth concepts.

(72)

STRUGARU, G., "A microprogrammed unit for computer systems", *IEEE Computer Group Repository R-69-95*; 1969.

"The paper examines the possibilities of utilizing the micro-subprograms...to reduce the number of micro-orders and to increase the flexibility..."

"In new computers there are introduced complex operations which result from the nonarithmetical applications of these, from the automation problems of programming, multiprogramming, etc."

"In this condition of complicating the microprogram, formal synthesis methods must be used and the microprogram minimization becomes essential for design."

(73)

TUCKER, S. G., "Microprogram control for System/360", *IBM Systems Journal*, Vol. 6, No. 4; 1967, pp. 222-241.

Machine: IBM 360 Systems

"This paper describes the kind of microprogram control that has been used in several models of System/360. A microprogramming language, as well as some of the main techniques used in "assembling" and testing microprograms, are discussed. Applications of microprogram control to the design of emulators, to compatibility features, and to special modifications are summarized."

(From the paper)

(74)

VALASSIS, J. G., "Modular computer design with picoprogrammed control", *AFIPS Conference Proceedings*; FJCC 1967.

"This paper describes the use of a multiaperture ferrite device called MYRA, and the use of picoprogramming to build a 16-bit, one microsecond integrated circuit computer. Picoprogramming is described as conceptually comparable to microprogramming but without the need for decoding logic. The MYRA provides the means for picoprogramming.

Organization of the computer, implementation of a microinstruction, operation of the MYRA, and the advantages and disadvantages of the technique are presented.

The paper is complete and well written."

H. Marks, Canoga Park, California,
Computing Reviews, Vol. 9, No. 6;
June 1968, #14,585.

(75)

VAN DER POEL, W. L., "Micro-programming and trickology", *Digital Information Processors; Selected Articles on Problems of Information Processing*; ed. Walter Hoffman; Interscience Publishers, a division of John Wiley and Sons, Inc.; New York, 1962, pp. 269-311.

Machine: ZEBRA

This article discusses the practical limitations of the dissection of macro-instructions into more elementary operations. The ZEBRA computer is used to test this dissection.

The argument that construction of the micro-code is too laborious for the programmer (van der Poel's concept of microprogramming is slightly different from Professor Wilkes' -- he sees it as an easily adjustable elementary machine language) is outweighed by its efficiency, especially in the area of automatic programming.

Mr. van der Poel also defines two concepts: trickology and underwater programming. The generation of pieces of coding in fast registers which are subsequently executed is called underwater programming. Considerable ingenuity is often required to devise the macro-instructions, giving rise to the name trickology for the art of using this tricky programming.

Three-fourths of this article is a description of ZEBRA. The examples and micro-code used are also from this computer. The article may be of some interest to those designing micromachines, or those actively (!) concerned with coding for the ZEBRA computer.

(76)

VANDLING, G. C. and WALDECKER, D. E., "The Microprogram Control Technique for Digital Logic Design", *Computer Design*, Vol. 8, No. 8; August 1969, pp. 44-51.

"Basic concepts of microprogrammed control design, representation of current design practice, and selected design variations are examined as sources of informative value to the logic designer."

(From the paper)

(77)

VOLLBRECHT, J. R., "MIDAS: Microprogramming Design Aid System for RCA Spectra 70/45", Presented at ACM Workshop on Microprogramming, October 7, 1968.

Machine: RCA Spectra 70/45

MIDAS, the Microprogramming Design Aid System, describes a program which performs much of the routine work necessary in developing a useful, effective microprogram.

Designing a microprogram requires "that a set of micro-orders be combined into a set of elementary operations in such a way that 1) the required function is in fact performed, and 2) the execution time is minimized." MIDAS allows the programmer to concentrate on this creative task by permitting him to "see" his microprogram and check it by a simulation process.

MIDAS includes an assembler, a simulator, a program to punch cards by which the read-only memory can be built, and a system to aid in debugging.

(78)

WEBER, H., "A microprogrammed implementation of EULER on IBM System/360 Model 30", *Communications of the ACM*, Vol. 10, No. 9; September 1967, pp. 549-558.

Machine: IBM System/360 Model 30

This article describes an experimental processing system for a higher level language, EULER. The language is processed in two stages. First a microprogrammed compiler translates it into a Reverse Polish String Language. Then this language is interpreted by a microprogram. The experiment was undertaken to show that microprogramming can be used to create more efficient systems than conventional programming systems. This system is predicted to be at least ten times faster than the execution of an ordinary (efficient) machine code compilation.

(79)

WEBER, H., "Microprogramming", *International Summer School on Fundamental Aspects and Current Developments in Computer Science*; August 1969.

In this course, Mr. Weber introduces the microprogram level language by analogy - as ALGOL is to machine language, so machine language is to a microlanguage.

Using several of the System 360 micromachines as examples, Mr. Weber weaves through the classical and later definitions of microprogramming, the types of read-only memory that are used and the possibilities of read-write memories. The 360/30 (2030) and 360/20 Submod 5 (2020-5) are examined in detail.

He also points out that "...a simulator, an interpreter, and the machine hardware performed exactly the same function, namely, to interpret string code. Why was it necessary to invent different names for simulator and interpreter?" "...emulators were announced. Again, a new word for an interpreting mechanism!"

Also, "the ideal processing system for a higher level language would therefore consist of a translator and an interpreter. Both the translator and the interpreter can be written in a micro-program language."

(80)

WILKES, M. V., "The best way to design an automatic calculating machine", *Manchester University Computer Inaugural Conference*, Manchester, England; July 1951, pp. 16-18.

This paper contains the original explanation and definition of the term microprogramming. A micro-operation is one of a series of steps. These steps may include transfers from the store to control or arithmetical registers and transfers from one register to another. The micro-operations can be generated by pulses. These pulses enter a decoding tree, and then are routed to two matrices, one of which executes an instruction, the other sets up the next instruction to be performed.

When introduced, microprogramming's main advantage seemed to Wilkes to be the increase in the degree of reliability in a machine. Reliability had a direct relationship to 1) the amount of equipment of a computer, 2) its complexity, and 3) the degree of repetition of units. Since microprograms can perform many operations with the same set of logic circuitry, it would appear that microprogramming did increase the reliability of a machine.

Since this was the original paper on the subject, the problems of programmer instruction, design, and time consumed are not discussed in detail.

(81)

WILKES, M. V., "Microprogramming", *Proceedings of the EJCC 1959*, pp. 18-20.

This article, written in 1958, gives a very brief outline on the development of microprogramming, including its advantages and potential. A microprogram is defined as "the list of suboperations or micro-operations required to execute all the orders in that order code".

The one outstanding advantage of microprograms is that changes in the order code can be incorporated at a very late date in the building of a computer.

The other new concept of the erasable or "scratch-pad" memory which allows the programmer to invent his own order code is also mentioned.

(82)

WILKES, M. V., "The growth of interest in microprogramming", ACM Workshop on Microprogramming, sponsored by the Association for Computing Machinery in cooperation with The MITRE Corporation, October 7-8, 1968.

Tracing the evolution of microprogramming from its conception in 1951 to the present, M. V. Wilkes examines its original and current uses and theorizes on further potential development.

A helpful explanation of the two current and different definitions of microprogramming are included. The concept of stored logic (a read-write control memory), a technique which allows the programmer to create his own microprograms, is also mentioned.

A history of microprogrammed computers is compiled, with brief comments provided on each referenced article.

Tucker's reasons for the use of microprogramming are summarized. It provides: 1) an economical way for smaller computers to have large instruction sets, 2) an aid to efficient maintenance, 3) the possibility of emulation, and 4) a gain in flexibility for both the programmer and designer.

The current state of the art is discussed in terms of further developments in emulation, software support, and the interpretation of higher-level programming languages.

(83)

WILKES, M. V., "The growth of interest in microprogramming: A literature survey", *Computing Surveys*, Vol. 1, No. 3; September 1969, pp. 139-145.

"The literature is surveyed beginning with the first paper published in 1951. At that time microprogramming was proposed primarily as a means for designing the control unit of an otherwise conventional digital computer, although the possible use of a read/write control memory was noted. The survey reveals the way in which interest has successively developed in the following aspects of the subject: stored logic, the application of microprogramming to the design of a range of computers, emulation, microprogramming in support of software, and read/write control memories. The bibliography includes 55 papers."

(From the paper)

(84)

WILKES, M. V. and STRINGER, J. B., "Microprogramming and the design of the control circuits in an electronic digital computer", *Proceedings of the Cambridge Philosophical Society*, Vol. 49, Part 2; 1953, pp. 230-238.

This article, dated November 18, 1952, describes in clear, concise terminology the structure and advantages of a microprogrammed computer.

Building on his original paper which introduced the concept of microprogramming in 1951, Professor Wilkes defines microprogramming as a method of designing the control circuits of a machine to be logical and yet enabling the order code to be alterable without ad hoc alterations to the circuits. The concept of a micro-control unit is discussed. Branching within the microprogram is shown to be accomplished by the setting of two flip-flops.

SECTION III

KMIC INDEX LISTING

DATE 10/23/69
TIME 01:42:03

INDEX

KEYWORD AND TEXT

25	ENTRIES WERE SUPPRESSED FOR KEYWORD -----> E	
5	ENTRIES WERE SUPPRESSED FOR KEYWORD -----> -	
1	ENTRIES WERE SUPPRESSED FOR KEYWORD -----> --	
177	ENTRIES WERE SUPPRESSED FOR KEYWORD -----> "	
34	ENTRIES WERE SUPPRESSED FOR KEYWORD -----> A	
52	MCGEE, W & PETERSON, H " MICROPROGRAMMING FOR DATA ACQUISITION AND CONTROL " 1965	
15	BREUER, M " ADAPTIVE COMPUTERS " 1967	
77	VOLLBRECHT, J " MIDAS: MICROPROGRAMMING DESIGN AID SYSTEM FOR RCA SPECTRA 70/45 " 1968	
	7 ENTRIES WERE SUPPRESSED FOR KEYWORD -----> AL	
	ACCESS TIME-SHARING SYSTEM "	
1	ALLEN-BABCOCK COMPUTING, INC " INFORMATION OUTLINE: REMOTE-A	
2	ALLEN, M; PEARCEY, T; PENNY, J ET AL " CIRCUITS, ECONOMICAL MULTI	
3	AMDAHL, L " MICROPROGRAMMING AND STORED LOGIC " 1964	
33	JTER WITH A PHOTOTRANSISTER FIXED MEMORY " 1962 HAGIWARA, H; AMO, K ET AL " THE KT PILOT COMPUTER - A MICROPROGRAMMED COMP	
	4 ENTRIES WERE SUPPRESSED FOR KEYWORD -----> AN	
	8 ENTRIES WERE SUPPRESSED FOR KEYWORD -----> AND	
	HILL, R " STORED LOGIC PROGRAMMING AND APPLICATIONS " 1964	
	HEELITZ, H; LEVY, S; LINHARDT, R & MILLER, H " SYSTEM ARCHITECTURE FOR LARGE SCALE INTEGRATION " 1967	
	LEVY, L " STATE OF THE ART OF MICROPROGRAMMING " 1965	
1958	MORRISON, G " INVESTIGATION OF MICRO-PROGRAMMING AS A TECHNIQUE IN THE CONTROL OF DIGITAL COMPUTING SYSTEMS "	
	CONTI, C; GIBSON, D & PITKOWSKY, S " STRUCTURAL ASPECTS OF THE SYSTEM/360, MODEL 85 " 1968	
	MOERNES, J ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1967	
	HELLENBAUM, L ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968	
	CALHOUN, R " DIAGNOSTICS AT THE MICROPROGRAMMING LEVEL " 1969	
	WILKES, M " THE BEST WAY TO DESIGN AN AUTOMATIC CALCULATING MACHINE " 1951	
	JACKINGHAM, B; CARTER, W; CRAWFORD, W & NOWELL, G " THE CONTROLS AUTOMATION SYSTEM " 1965	
1967	HAILEY, S " FASTER COMPUTER CONTROL WITH A READ-ONLY MEMORY "	
	BASHKOW, T; SASSON, A & KRONFELD, A " SYSTEM DESIGN OF A FORTRA	
	HAZERQUE, G; FERRIE, J & HUGOT, P " MICROMACHINE UNIVERSAL STRU	
	BECK, L & KEELER, F " THE C-8401 DATA PROCESSOR " 1964	
	HEELITZ, H; LEVY, S; LINHARDT, R & MILLER, H " SYSTEM ARCHITECTU	
	BENJAMIN, R " THE SPECTRA 70/45 EMULATOR FOR THE RCA 301 " 19	
	BEST WAY TO DESIGN AN AUTOMATIC CALCULATING MACHINE " 1951	
	BILLING, H & HOPMANN, W " MICROPROGRAMM-STEUERWERK " 1955	
	BLANKENBAKER, J " LOGICALLY MICRO-PROGRAMMED COMPUTERS " 1960	
	WILKES, M " THE	
	RE FOR LARGE SCALE INTEGRATION " 1967	
55		

INDEX

52

KWIC INDEX LISTING

DATE 10/23/69
TIME 01:42:03

TEXT

KEYWORD AND TEXT

INDEX

ING AND THE DESIGN OF THE CONTROL CIRCUITS IN AN ELECTRONIC DIGITAL COMPUTER " 1953WILKES,M & STRINGER,J " MICROPROGRAMM 84
GRAMMING THE SPECTRA 70/35 " 1966 N A VARIABLE STRUCTURE DIGITAL COMPUTER " 1962 CAMPBELL,C & NEILSON,D " MICROPRO 19
YANDOV-YAROSLAVSKY,L & TIMOFEEV,A " MICROPROGRAM CONTROL FOR DIGITAL COMPUTERS " 1962 EMEL 26
ATION OF MICRO-PROGRAMMING AS A TECHNIQUE IN THE CONTROL OF DIGITAL COMPUTING SYSTEMS " 1958 MORRISON,G " INVESTIG 57
NG,C & WALDECKER,D " THE MICROPROGRAM CONTROL TECHNIQUE FOR DIGITAL LOGIC DESIGN " 1969 76
HILBOURNE,A & PUGHIRE,J " A SMALL COMPUTER FOR THE DIRECT PROCESSING OF FORTRAN STATEMENTS " 1965 54
" A MICROPROGRAMMED COMMUNICATION CONTROL UNIT, THE TOSBAC OPLER,A " NEW DIRECTIONS IN SOFTWARE 1960-1966 " 1966 59
DN-231 " 1968 MATSUSHITA,S 47
DREYER,L " PRINCIPLES OF A TWO-LEVEL MEMORY COMPUTER " 1968 25

1963 ROSE,G; SANDERSON,J ET AL " CIRCUITS, ECONOMICAL MULTIPROGRAM COMPUTER WITH MICROPROGRAM CONTROL " 2
1963 ALLEN,M; PEARCEY,T; PENNY,J ET AL " CIRCUITS, ECONOMICAL MULTIPROGRAM COMPUTER WITH MICROPROGRAM CONTROL " 2
CROPPROGRAMMING AND THE DESIGN OF THE CONTROL CIRCUITS IN AN ELECTRONIC DIGITAL COMPUTER " 1953WILKES,M & STRINGER,J " MI 84
OF A GENERAL-PURPOSE MICROPROGRAM-CONTROLLED COMPUTER WITH ELEMENTARY STRUCTURE " 1960 KAMPE,T " THE DESIGN 42
FOR DIGITAL COMPUTERS " 1962 EMEYANOV-YAROSLAVSKY,L & TIMOFEEV,A " MICROPROGRAM CONTROL 26
BENJAMIN,R " THE SPECTRA 70/45 EMULATOR FOR THE RCA 301 " 1965 9
GREEN,J " MICROPROGRAMMING, EMULATORS AND PROGRAMMING LANGUAGES " 1966 31

7 ENTRIES WERE SUPPRESSED FOR KEYWORD -----> ET

WEBER,H " A MICROPROGRAMMED IMPLEMENTATION OF EULER ON IBM SYSTEM/360 MODEL 30 " 1967 78
HARRAND,Y " EVOLUTION OF MICROPROGRAMMING CONCEPTS " 1963 34
MCGEE,M & PETERSON,H " MICROPROGRAM CONTROL FOR THE EXPERIMENTAL SCIENCES " 1965 51
REBOULOT,M " EXTRACTEUR MICROPROGRAMME D' INFORMATION RADAR " 1968 62

BAILEY,S " FASTER COMPUTER CONTROL WITH A READ-ONLY MEMORY " 1967 4
MCCORMACK,M; SCHANSHAN,T & WOMACK,K " 1401 COMPATIBILITY FEATURE ON THE IBM SYSTEM/360 MODEL 30 " 1965 49
ORIENTED TO COMPUTERS SIMULATION " 1968 BAZERQUE,G; FERIE,J & HUGOT,P " MICROMACHINE UNIVERSAL STRUCTURE STUDY 6
OMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962 YAMAUCHI,H ET AL " THE KT PILOT C 33
OMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962 MATSUSHITA,S ET AL " THE KT PILOT C 33
OMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962 HAGIWARA,H; AMO,K ET AL " THE KT PILOT C 33
FLYNN,M & MACLAREN,M " MICROPROGRAMMING REVISITED " 1967 27

16 ENTRIES WERE SUPPRESSED FOR KEYWORD -----> FOR

BASHKOV,T; SASSON,A & KRONFELD,A " SYSTEM DESIGN OF A FORTRAN MACHINE " 1967 5
PUGHIRE,J " A SMALL COMPUTER FOR THE DIRECT PROCESSING OF FORTRAN STATEMENTS " 1965 MELBOURNE,A 54
RAKOCZI,L " THE COMPUTER-WITHIN-A-COMPUTER A FOURTH GENERATION CONCEPT " 1969 61
OPLER,A " FOURTH-GENERATION SOFTWARE: THE REALIGNMENT " 1967 58

TARY STRUCTURE " 1960 KAMPE,T " THE DESIGN OF A GENERAL-PURPOSE MICROPROGRAM-CONTROLLED COMPUTER WITH ELEMEN 42
RAKOCZI,L " THE COMPUTER-WITHIN-A-COMPUTER A FOURTH GENERATION CONCEPT " 1969 61
963 GERACE,G " MICROPROGRAMMED CONTROL FOR COMPUTING SYSTEMS " 1 28
D, MODEL 85 " 1968 GIBSON,O & PITKOWSKY,S " STRUCTURAL ASPECTS OF THE SYSTEM/36 23
CONTI,C; GLANTZ,H " A NOTE ON MICROPROGRAMMING " 1956 29
ROSE,G " INTERGRAPHIC, A MICROPROGRAMMED GRAPHICAL-INTERFACE COMPUTER " 1967 63
MED CONTROL UNITS " 1962 GRASELLI,A " THE DESIGN OF PROGRAM-MODIFIABLE MICRO-PROGRAM 30
GES " 1966 GREENBERG,M & WEGBREIT,E " A MICROPROGRAMMED COMPUTER " 1968 31
" 1969 WILKES,M " THE GROWTH OF INTEREST IN MICROPROGRAMMING " 1968 82
WILKES,M " THE GROWTH OF INTEREST IN MICROPROGRAMMING: A LITERATURE SURVEY 83

GRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962 HAGIWARA,H; AMO,K ET AL " THE KT PILOT COMPUTER - A MICROPRO 33

KWIC INDEX LISTING

DATE 10/23/69
TIME 01:42:03

INDEX

KEYWORD AND INDEX

TEXT

OLUTION, THE VARIABLE INSTRUCTION COMPUTER " 1966 SPENCE, A " 69
ROSEN, S " 64
HARDWARE DESIGN REFLECTING SOFTWARE REQUIREMENTS " 1968
HARRAND, Y " EVOLUTION OF MICROPROGRAMMING CONCEPTS " 1963
HAWRYSZKIEWICZ, I " MICROPROGRAMMED CONTROL IN PROBLEM-ORIENTED LANGUAGES " 1967 34
HECKELMAN, R " SELF-CHECKING REDUNDANT MICROPROGRAMS " 1966 35
HELLERMAN, L ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968 36
HEGGITT, J " A CHARACTER COMPUTER FOR 37
CHU, Y " A HIGH LEVEL LANGUAGE INTERPRETATION " 1964 53
HILL, R " STORED LOGIC PROGRAMMING AND APPLICATIONS " 1964 20
HOERNES, G ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968 37
BOUTHELL, E & HOSKINSON, E " THE LOGICAL ORGANIZATION OF THE PB 44D MICROPROGRAMMABLE COMPUTER " 1963 10
BAZERQUE, G; FERRIE, J & HUGOT, P " MICROMACHINE UNIVERSAL STRUCTURE STUDY ORIENTED TO COMPUTERS SIMULATION " 1968 6
SCHANSMAN, T & WOMACK, K " 1401 COMPATIBILITY FEATURE ON THE IBM SYSTEM/360 MODEL 30 " 1965 39
WEBER, H " A MICROPROGRAMMED IMPLEMENTATION OF EULER ON IBM SYSTEM/360 MODEL 30 " 1967 49
STEVENS, W " THE STRUCTURE OF SYSTEM/360; PART II - SYSTEM IMPLEMENTATIONS " 1964 78
WEBER, H " A MICROPROGRAMMED IMPLEMENTATION OF EULER ON IBM SYSTEM/360 MODEL 30 " 1967 70
STEVENS, W " THE STRUCTURE OF SYSTEM/360; PART II - SYSTEM IMPLEMENTATIONS " 1964 70
HELLERMAN, L ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968 37
HOERNES, G ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968 37
PATZER, W & VANDLING, G " SYSTEMS IMPLICATIONS OF MICROPROGRAMMING " 1967 60
ALLEN-BABCOCK COMPUTING, INC " INFORMATION OUTLINE: REMOTE-ACCESS TIME-SHARING SYSTEM " 1
ALLEN-BABCOCK COMPUTING, INC " INFORMATION OUTLINE: REMOTE-ACCESS TIME-SHARING SYSTEM " 1
REBOULOT, M " EXTRACTEUR MICROPROGRAMME D'INFORMATION RADAR " 1968 62
INGALLS, R " LOGICAL DESIGN OF A MICROPROGRAMMED SPECIAL-PURPOSE COMPUTER " 1966 40
MILLER, E " RELIABILITY ASPECTS OF THE VARIABLE INSTRUCTION COMPUTER " 1966 SPENCE, A " HARDWARE AND SOFTWARE INTERACTIONS: A MACHINE ORGANIZATION SOLUTION, THE VARIABLE INSTRUCTION COMPUTER " 1967 69
LAWSON, H " PROGRAMMING-LANGUAGE-ORIENTED INSTRUCTION STREAMS " 1968 56
LINHARDT, R & MILLER, H " SYSTEM ARCHITECTURE FOR LARGE SCALE INTERACTION " 1967 44
WILKES, M " THE GROWTH OF INTEREST IN MICROPROGRAMMING " 1968 8
WILKES, M " THE GROWTH OF INTEREST IN MICROPROGRAMMING " 1968 8
WILKES, M " THE GROWTH OF INTEREST IN MICROPROGRAMMING: A LITERATURE SURVEY " 1969 69
WILKES, M " THE GROWTH OF INTEREST IN MICROPROGRAMMING: A LITERATURE SURVEY " 1969 83
ROSE, G " INTERGRAPHIC, A MICROPROGRAMMED GRAPHICAL-INTERFACE COMPUTER DECISION SYSTEMS, INC " INTERIM STATUS REPORT NO. 1 " 1968 24
MEGGITT, J " A CHARACTER COMPUTER FOR HIGH LEVEL LANGUAGE INTERPRETATION " 1964 53
MORRISON, G " INVESTIGATION OF MICRO-PROGRAMMING AS A TECHNIQUE IN THE CONTROL OF DIGITAL COMPUTING SYSTEMS " 1958 57
IVERSON, K " A PROGRAMMING LANGUAGE " 1962 41
KAMPE, T " THE DESIGN OF A GENERAL-PURPOSE MICROPROGRAM-CONTROLLED COMPUTER WITH ELEMENTARY STRUCTURE " 1960 42
BECK, L & KEELER, F " THE C-8401 DATA PROCESSOR " 1964 7
BORSCHKE, V; VASILEVSKIY, P & KHEIM, S " A PROGRAMMING MICRO-LANGUAGE FOR URAL-4 " 1966 12
KHEIM, S & SCHWARTZ, S " MODEL 4200 - B200 READ-ONLY MEMORY CONTROL LOGIC " 1968 43
BASHKON, T; SASSON, A & KRONFELD, A " SYSTEM DESIGN OF A FORTRAN MACHINE " 1967 5
MATSUHITA, S ET AL " THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTO TRANSISTOR FIXED MEMORY " 1962 33
YAMAUCHI, H ET AL " THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTO TRANSISTOR FIXED MEMORY " 1962 33
HAGIMAWA, H; AMO, K ET AL " THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTO TRANSISTOR FIXED MEMORY " 1962 33
IVERSON, K " A PROGRAMMING LANGUAGE " 1962 41

B ENTRIES WERE SUPPRESSED FOR KEYWORD ----- IN

DATE
TIME 10/23/69
01:42:03

INDEX

KEYWORD AND INDEX

INDEX

- CHU, Y " A HIGH-ORDER COMPUTER FOR HIGH LEVEL LANGUAGE INTERPRETATION " 1964
 GREEN, J " MICROPROGRAMMING, EMULATORS AND PROGRAMMING LANGUAGES " 1966
 PYSZKIEWICZ, J " MICROPROGRAMMED CONTROL IN PROBLEM-ORIENTED LANGUAGES " 1967
 LEV, S; LINHARDT, R & MILLER, H " SYSTEM ARCHITECTURE FOR LARGE SCALE INTEGRATION " 1967
 CALHOUN, R " DIAGNOSTICS AT THE MICROPROGRAMMING LEVEL " 1969
 MEGGITT, J " A CHARACTER COMPUTER FOR HIGH LEVEL LANGUAGE INTERPRETATION " 1964
 LEV, L " STATE OF THE ART OF MICROPROGRAMMING " 1965
 LEV, S; LINHARDT, R & MILLER, H " SYSTEM ARCHITECTURE FOR LARGE SCALE INTEGRATION " 1967
 LEV, S; LINHARDT, R & MILLER, H " SYSTEM ARCHITECTURE FOR LARGE SCALE INTEGRATION " 1967
 WILKES, M " THE GROWTH OF INTEREST IN MICROPROGRAMMING: A LITERATURE SURVEY " 1969
 LIVERANCE, H " A SYSTEM/360 PROFILE -- READ-ONLY STORAGE " 1966
 AMDAHL, L " MICROPROGRAMMING AND STORED LOGIC " 1964
 S & SCHWARTZ, S " MODEL 4200 - 8200 READ-ONLY MEMORY CONTROL LOGIC " 1968
 SEMARNE, H & PETERSON, H " A STORED LOGIC COMPUTER " 1961
 SEMARNE, H & MCGEE, M " STORED LOGIC COMPUTING " 1961
 ALDECKER, D " THE MICROPROGRAM CONTROL TECHNIQUE FOR DIGITAL LOGIC DESIGN " 1969
 HILL, R " STORED LOGIC PROGRAMMING AND APPLICATIONS " 1964
 INCALLS, R " LOGICAL DESIGN OF A MICROPROGRAMMED SPECIAL-PURPOSE COMPUTER " 1966
 BOUTWELL, E & HOSKINSON, E " THE LOGICAL ORGANIZATION OF THE P8 440 MICROPROGRAMMABLE COMPUTER " 1963
 BLANKENBAKER, J " LOGICALLY MICRO-PROGRAMMED COMPUTERS " 1960
 WILKES, M " THE BEST WAY TO DESIGN AN AUTOMATIC CALCULATING MACHINE " 1951
 SHKOW, T; SASSON, A & KRUMHOLTZ, A " SYSTEM DESIGN OF A FORTRAN MACHINE " 1967
 JET, " 1966 SPENCE, A " HARDWARE AND SOFTWARE INTERACTIONS: A MACHINE ORGANIZATION SOLUTION, THE VARIABLE INSTRUCTION COMP FLYN, P & MACLAUREN, M " MICROPROGRAMMED COMMUNICATION CONTROL UNIT, MATSUSHITA, S ET AL " THE KT PILOT COMPUTER - A MICROPROGRAM 47
 THE TOSBAC DN-231 " 1968
 ED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962
 TURE (IN THE IBM SYSTEM/360 MODEL 30 " 1965
 TAL SCIENCES " 1965
 AND CONTROL " 1965
 SEMARNE, H & MCGEE, M " STORED LOGIC COMPUTING " 1961
 CONROY, E & HEADE, R " A MICROINSTRUCTION SYSTEM " 1961
 MEGGITT, J " A CHARACTER COMPUTER FOR HIGH LEVEL LANGUAGE INTERPRETATION " 1964
 MELBOURNE, A & PUGH, J " A SMALL COMPUTER FOR THE DIRECT PR 54
 MEMORY " 1962
 YAMAUCHI, H ET AL " THE KT PILOT COMPUTER 33
 MEMORY " 1962
 MATSUSHITA, S ET AL " THE KT PILOT COMPUTER 33
 R - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962
 HAILEY, S " FASTER COMPUTER CONTROL WITH A READ-ONLY MEMORY " 1967
 DREYER, L " PRINCIPLES OF A TWO-LEVEL MEMORY COMPUTER " 1968
 KLEIN, S & SCHWARTZ, S " MODEL 4200 - 8200 READ-ONLY MEMORY CONTROL LOGIC " 1968
 MCGEE, M " MICRO-PROGRAMMING " 1957
 MCGEE, R " MICRO-PROGRAMMING " 1957
 METHUEN, J " MINIMIZING MICROPROGRAMS " 1967
 MICRO-PROGRAMMING FOR URAL-4 " 1966
 MICRO-PROGRAMMED COMPUTERS " 1960
 MICRO-PROGRAMMED CONTROL UNITS " 1962
 MICRO-PROGRAMMING AND TRICKLOGY " 1962
 MICRO-PROGRAMMING AS A TECHNIQUE IN THE CONTROL OF DIGITAL C 57
 MORRISON, G " INVESTIGATION OF MICRO-PROGRAMMING AS A TECHNIQUE IN THE CONTROL OF DIGITAL C 57
 CONROY, E & HEADE, R " A MICROINSTRUCTION SYSTEM " 1961

KWIC INDEX LISTING

DATE 10/23/69
TIME 01:42:03

INDEX

KEYWORD AND TEXT

INDEX

SIMULATION " 1968 BAZERQUE,G; FERRIE,J & HUGOT,P " MICRO MACHINE UNIVERSAL STRUCTURE STUDY ORIENTED TO COMPUTERS 6
 SON,J ET AL " CIRCUITS, ECONOMICAL MULTIPROGRAM COMPUTER WITH MICROPROGRAM CONTROL " 1963 ROSE,G; SANDER 2
 NNY,J ET AL " CIRCUITS, ECONOMICAL MULTIPROGRAM COMPUTER WITH MICROPROGRAM CONTROL " 1963 ALLEN,M; PEARCEY,T; PE 2
 EHELYANOV-YAROSLAVSKY,L & TIMOREEV,A " MICROPROGRAM CONTROL FOR DIGITAL COMPUTERS " 1962 26
 TUCKER,S " MICROPROGRAM CONTROL FOR SYSTEM/360 " 1967 73
 59 MCGEE,M & PETERSON,H " MICROPROGRAM CONTROL FOR THE EXPERIMENTAL SCIENCES " 1965 51
 VANDLING,G & WALDECKER,D " THE MICROPROGRAM CONTROL TECHNIQUE FOR DIGITAL LOGIC DESIGN " 19 76
 1960 KAMPE,T " THE DESIGN OF A GENERAL-PURPOSE MICROPROGRAM-CONTROLLED COMPUTER WITH ELEMENTARY STRUCTURE " 42
 LL,E & HOSKINSON,E " THE LOGICAL ORGANIZATION OF THE PB 440 MICROPROGRAMMABLE COMPUTER " 1963 BOUTWE 14
 REBOULOT,M " EXTRACTEUR MICROPROGRAMME D' INFORMATION RADAR " 1968 62
 1 " 1968 NATSUSHITA,S " A MICROPROGRAMMED COMMUNICATION CONTROL UNIT, THE TOSBAC ON-23 47
 GREENBERG,M & WEBREIT,E " A MICROPROGRAMMED COMPUTER " 1968 32
 MCCLURE,R & DAVIS,R " A SMALL MICROPROGRAMMED COMPUTER " 1969 48
 SCHLAEPPI,H " MICROPROGRAMMED COMPUTER CONTROL " 1963 65
 " 1962 HAGIWARA,H; ANDO,K ET AL " THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY 33
 " 1962 MATSUSHITA,S ET AL " THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY 33
 " 1962 YAMAUCHI,H ET AL " THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY 20
 CHU,Y " A HIGH-ORDER LANGUAGE FOR DESCRIBING GERACE,G " MICROPROGRAMMED CONTROL FOR COMPUTING SYSTEMS " 1963 28
 -JAWYSZKIENYCZ,I " MICROPROGRAMMED CONTROL IN PROBLEM-ORIENTED LANGUAGES " 1967 35
 ROSE,G " INTERGRAPHIC, A MICROPROGRAMMED GRAPHICAL-INTERFACE COMPUTER " 1967 63
 INCALLS,R " LOGICAL DESIGN OF A MICROPROGRAMMED IMPLEMENTATION OF EULER ON IBM SYSTEM/360 MO 78
 STRUGARU,C " A MICROPROGRAMMED SPECIAL-PURPOSE COMPUTER " 1966 40
 GLANTZ,H " A NOTE ON MICROPROGRAMMING " 1956 29
 WILKES,M " MICROPROGRAMMING " 1959 81
 CONROY,E " MICROPROGRAMMING " 1961 21
 LEVY,L " STATE OF THE ART OF MICROPROGRAMMING " 1965 45
 PATZER,M & VANDLING,G " SYSTEMS IMPLICATIONS OF MICROPROGRAMMING " 1967 60
 WILKES,M " THE GROWTH OF INTEREST IN MICROPROGRAMMING " 1968 79
 WEBER,H " MICROPROGRAMMING AND STORED LOGIC " 1964 3
 AMOHL,L " MICROPROGRAMMING AND THE CHOICE OF ORDER CODE " 1953 71
 STRINGER,J " MICROPROGRAMMING AND THE DESIGN OF THE CONTROL CIRCUITS IN A 84
 HARRAND,Y " EVOLUTION OF MICROPROGRAMMING CONCEPTS " 1963 34
 968 VOLBRECHT,J " MIOAS: MICROPROGRAMMING DESIGN AID SYSTEM FOR RCA SPECTRA 70/45 " 1 77
 MCGEE,M & PETERSON,H " MICROPROGRAMMING FOR DATA ACQUISITION AND CONTROL " 1965 52
 CALHOUN,R " DIAGNOSTICS AT THE MICROPROGRAMMING LEVEL " 1969 18
 IBM " DATA PROCESSING: MICROPROGRAMMING PI COURSE " 1967 39
 FLYNN,M & MACAREN,M " MICROPROGRAMMING REVISITED " 1967 27
 WILKES,M " THE GROWTH OF INTEREST IN MICROPROGRAMMING THE SPECTRA 70/35 " 1966 N A VARIABLE ST 19
 HECKELMAN,F " SELF-CHECKING REDUNDANT MICROPROGRAMS " 1966 31
 SLOBODYANYUK,T " A METHOD OF MINIMIZING MICROPROGRAMS " 1966 83
 VOLBRECHT,J " MIOAS: MICROPROGRAMMING DESIGN AID SYSTEM FOR RCA SPECTRA 70 36
 BILLING,H & HOPMANN,M " MICROPROGRAM-STEUERWERK " 1955 68
 745 " 1968 MILLER,E " RELIABILITY ASPECTS OF THE VARIABLE INSTRUCTION C 77
 COMPUTER " 1967 MICROPROGRAMMING: A LITERATURE SURVEY " 1969 10
 1967 HEELITZ,H; LEVY,S; LINHARDT,R & SLOBODYANYUK,T " A METHOD OF MINIMIZING MICROPROGRAMS " 1967 56
 WOMACK,K " 1401 COMPATIBILITY FEATURE ON THE IBM SYSTEM/360 MODEL 30 " 1965 8
 A MICROPROGRAMMED IMPLEMENTATION OF EULER ON IBM SYSTEM/360 MODEL 30 " 1967 68
 KLEIN,S & SCHWARTZ,S " MODEL 4200 - 8200 READ-ONLY MEMORY CONTROL LOGIC " 1968 49
 43 WEBER,H " MCCORMACK,M; SCHANSMAN,T & 78

KWIC INDEX LISTING

DATE 10/23/69
TIME 01:42:03

INDEX

KEYWORD AND TEXT

SON,D & PITKOWSKY,S " STRUCTURAL ASPECTS OF THE SYSTEM/360, MODEL 85 " 1968 CONTI,C; GIB 23
VALASSIS,J " MODULAR COMPUTER DESIGN WITH PICOPROGRAMMED CONTROL " 1967 74
JE IN THE CONTROL OF DIGITAL COMPUTING SYSTEMS " 1958 MORRISON,G " INVESTIGATION OF MICRO-PROGRAMMING AS A TECHNIQ 57
ALLEN,M; PEARCEY,T; PENNY,J ET AL " CIRRUS, ECONOMICAL MULTIPROGRAM COMPUTER WITH MICROPROGRAM CONTROL " 1963 2
RUSE,G; SANDERSON,J ET AL " CIRRUS, ECONOMICAL MULTIPROGRAM COMPUTER WITH MICROPROGRAM CONTROL " 1963 2
G NEILSON,D " MICROPROGRAMMING THE SPECTRA 70/35 " 1966 N A VARIABLE STRUCTURE DIGITAL COMPUTER " 1962 CAMPBELL,C 19
VARIABLE STRUCTURE DIGITAL COMPUTER " 1962 CAMPBELL,C & NEILSON,D " MICROPROGRAMMING THE SPECTRA 70/35 " 1966 N A 19
DECISION SYSTEMS,INC " INTERIM STATUS REPORT NO. I " 1968 OPLER,A " NEW DIRECTIONS IN SOFTWARE 1960-1966 " 1966 59
GLANTZ,H " A NOTE ON MICROPROGRAMMING " 1956 24
BUCKINGHAM,B; CARTER,W; CRAWFORD,W & NOWELL,G " THE CONTROLS AUTOMATION SYSTEM " 1965 29
22 ENTRIES WERE SUPPRESSED FOR KEYWORD -----> OF 17

3 ENTRIES WERE SUPPRESSED FOR KEYWORD -----> ON

STRINGER,J " MICROPROGRAMMING AND THE CHOICE OF ORDER CODE " 1953 OPLER,A " FOURTH-GENERATION SOFTWARE: THE REALIGNMENT " 1967 58
BOUTWELL,E & HOSKINSON,E " THE LOGICAL ORGANIZATION OF THE PB 440 MICROPROGRAMMABLE COMPUTER " 1963 59
966SPENCE,A " HARDWARE AND SOFTWARE INTERACTIONS: A MACHINE ORGANIZATION SOLUTION, THE VARIABLE INSTRUCTION COMPUTER " 1 71
FERRIE,J & HUGOT,P " MICRO MACHINE UNIVERSAL STRUCTURE STUDY ORIENTED TO COMPUTERS SIMULATION " 1968 14
ALLEN-BABCOCK COMPUTING, INC " INFORMATION OUTLINE: REMOTE-ACCESS TIME-SHARING SYSTEM " 6
BAZARQUE,G; 1

STEVENS,M " THE STRUCTURE OF SYSTEM/360; PART II - SYSTEM IMPLEMENTATIONS " 1964 70
PATZER,M & VANDLING,G " SYSTEMS IMPLICATIONS OF MICROPROGRAM 60
PB 440 COMPUTER " 1964 13
BOUTWELL,E " THE LOGICAL ORGANIZATION OF THE PB 440 MICROPROGRAMMABLE COMPUTER " 1963 14

ALLEN,M; PEARCEY,T; PENNY,J ET AL " CIRRUS, ECONOMICAL MULTIPROGRAM C 2
4 MICROPROGRAM CONTROL " 1963 ALLEN,M; PEARCEY,T; PENNY,J ET AL " CIRRUS, ECONOMICAL MULTIPROGRAM COMPUTER WITH 2
CES " 1965 MCCEE,M & PETERSON,H " MICROPROGRAM CONTROL FOR THE EXPERIMENTAL SCIEN 51
" THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962 HAGIMARA,H; AMO,K ET AL 33
" THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962 YAMAUCHI,H ET AL 33
" THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962 MATSUSHITA,S ET AL 33

VALASSIS,J " MODULAR COMPUTER DESIGN WITH PICOPROGRAMMED CONTROL " 1967 74
SISTOR FIXED MEMORY " 1962 MATSUSHITA,S ET AL " A MICROPROGRAMMED COMPUTER WITH A PHOTOTRAN 33
SISTOR FIXED MEMORY " 1962 HAGIMARA,H; AMO,K ET AL " A MICROPROGRAMMED COMPUTER WITH A PHOTOTRAN 33
SISTOR FIXED MEMORY " 1962 YAMAUCHI,H ET AL " A MICROPROGRAMMED COMPUTER WITH A PHOTOTRAN 33
CONTI,C; GIBSON,D & PITKOWSKY,S " STRUCTURAL ASPECTS OF THE SYSTEM/360, MODEL 85 23
VAN DER POEL,W " MICRO-PROGRAMMING AND TRICKOLOGY " 1962 75
SEMARNE,H & PORTER,R " A STORED LOGIC COMPUTER " 1961 67
DREYER,L " PRINCIPLES OF A TWO-LEVEL MEMORY COMPUTER " 1968 25
BRILEY,B " MICROPROGRAMMING " 1965 16
HAWRYSZKIEWICZ,I " MICROPROGRAMMED CONTROL IN PROBLEM-ORIENTED LANGUAGES " 1967 35
MELBOURNE,A & PUGHIRE,J " A SMALL COMPUTER FOR THE DIRECT PROCESSING OF FORTRAN STATEMENTS " 1965 54
IBM " DATA PROCESSING: MICROPROGRAMMING PI COURSE " 39
BECK,L & KEELER,F " THE C-8401 DATA PROCESSOR " 1964 7
IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968 37

HJERNES,G ET AL " CONTROL STORAGE USE

KWIC INDEX LISTING

DATE 10/23/69
TIME 01:42:03

TEXT

KEYWORD AND TEXT

INDEX

IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968 HELLERMAN, L ET AL " CONTROL STORAGE USE 37
ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968 HOERNES, G 37
ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968 HELLERMAN, L 37
LIVERANCE, H " A SYSTEM/360 PROFILE -- READ-ONLY STORAGE " 1965 46
GRASSELLI, A " THE DESIGN OF PROGRAM-MODIFIABLE MICRO-PROGRAMMED CONTROL UNITS " 1962 30
HILL, R " STORED LOGIC PROGRAMMING AND APPLICATIONS " 1964 38
IVERSON, K " A PROGRAMMING LANGUAGE " 1962 41
GREEN, J " MICROPROGRAMMING, EMULATORS AND PROGRAMMING LANGUAGES " 1966 31
BORSCHKEV, V; VASILEVSKIY, P & KHOMYAKOV, M " A PROGRAMMING MICRO-LANGUAGE FOR URAL-4 " 1966 12
LAWSON, H " PROGRAMMING-LANGUAGE-ORIENTED INSTRUCTION STREAMS " 1968 44
MELBOURNE, A & PUGMIRE, J " A SMALL COMPUTER FOR THE DIRECT PROCESSING OF FORTRAN STATEMENTS " 1965 54
REBOULOT, M " EXTRACTEUR MICROPRUGRAMME D' INFORMATION RADAR " 1968 62
ION CONCEPT " 1969 61
OLLBRECHT, J " MIUAS: MICROPROGRAMMING DESIGN AID SYSTEM FOR RCA SPECTRA 70/45 " 1968 77
BENJAMIN, R " THE SPECTRA 70/45 EMULATOR FOR THE RCA 301 " 1965 9
BAILEY, S " FASTER COMPUTER CONTROL WITH A READ-ONLY MEMORY " 1967 4
KLEIN, S & SCHWARTZ, S " MODEL 4200 - 8200 READ-ONLY MEMORY CONTROL LOGIC " 1968 43
LIVERANCE, H " A SYSTEM/360 PROFILE -- READ-ONLY STORAGE " 1965 46
UPLER, A " FOURTH-GENERATION SOFTWARE: THE REALIGNMENT " 1967 58
HECKELMAN, R " SELF-CHECKING REDUNDANT MICROPROGRAMS " 1966 62
ROSEN, S " HARDWARE DESIGN REFLECTING SOFTWARE REQUIREMENTS " 1968 36
MILLER, E " RELIABILITY ASPECTS OF THE VARIABLE INSTRUCTION COMPUTER " 1 64
ALLEN-BABCOCK COMPUTING, INC " INFORMATION OUTLINE: REMOTE-ACCESS TIME-SHARING SYSTEM " 1 56
ROSEN, S " HARDWARE DESIGN REFLECTING SOFTWARE REQUIREMENTS " 1968 1
FLYNN, M & MACLAREN, M " MICROPROGRAMMING REVISED " 1967 24
E COMPUTER " 1967 27
COMPUTER WITH MICROPROGRAM CONTROL " 1963 63
1968 64
WITH MICROPROGRAM CONTROL " 1963 2
1967 5
VY, S; LINHART, R & MILLER, H " SYSTEM ARCHITECTURE FOR LARGE SCALE INTEGRATION " 1967 8
MCCORRACK, M; SCHANSMAN, T & WOMACK, K " 1401 COMPATIBILITY FEATURE ON THE IBM SYSTEM/360 MODEL 30 " 1965 49
C " 1968 65
M & PETERSUN, H " MICROPROGRAM CONTROL FOR THE EXPERIMENTAL SCIENCES " 1965 43
HECKELMAN, R " SELF-CHECKING REDUNDANT MICROPROGRAMS " 1966 51
SEMARNE, H & MCGEE, W " STORED LOGIC COMPUTING " 1961 36
SIMULATION " 1968 66
SLOBODYANYUK, T " A METHOD OF MINIMIZING MICROPROGRAMS " 1967 67
BIZERQUE, G; FERRIE, J & HUGOT, P " M 6
TS " 1965 68
MELBOURNE, A & PUGMIRE, J " A SMALL COMPUTER FOR THE DIRECT PROCESSING OF FORTRAN STATEMENTS 54
MCCLURE, R & DAVIS, R " A SMALL MICROPROGRAMMED COMPUTER " 1969 48
ROSEN, S " HARDWARE DESIGN REFLECTING SOFTWARE REQUIREMENTS " 1968 69
VARIABLE INSTRUCTION COMPUTER " 1966SPENCE, A " HARDWARE AND SOFTWARE INTERACTIONS: A MACHINE ORGANIZATION SOLUTION, THE 64
ROSEN, S " NEW DIRECTIONS IN SOFTWARE: THE REALIGNMENT " 1967 59
OPLER, A " FORTH-GENERATION SOFTWARE: THE REALIGNMENT " 1967 58
OPLER, A " FORTH-GENERATION SOFTWARE: THE REALIGNMENT " 1967 58
HARDWARE AND SOFTWARE INTERACTIONS: A MACHINE ORGANIZATION SOLUTION, THE VARIABLE INSTRUCTION COMPUTER " 1966SPENCE, A " 69
INGALLS, R " LOGICAL DESIGN OF A MICROPROGRAMMED SPECIAL-PURPOSE COMPUTER " 1966 40
TER " 1962 CAMPBELL, C & NEILSON, D " MICROPROGRAMMING THE SPECTRA 70/35 " 1966 N A VARIABLE STRUCTURE DIGITAL COMPU 19

KWIC INDEX LISTING

DATE
TIME

ID/23/69
DL:42:03

INDEX

KEYWORD AND TEXT

TEXT

RECHT, J " MIDAS: MICROPROGRAMMING DESIGN AID SYSTEM FOR RCA SPECTRA 70/45 " 1968	VOLLB	77
BENJAMIN, R " THE SPECTRA 70/45 EMULATOR FOR THE RCA 301 " 1965		9
ANIZATION SOLUTION, THE VARIABLE INSTRUCTION COMPUTER " 1966	SPECTRA, A " HARDWARE AND SOFTWARE INTERACTIONS: A MACHINE ORG	69
LEVY, L " STATE OF THE ART OF MICROPROGRAMMING " 1965		45
E, J " A SMALL COMPUTER FOR THE DIRECT PROCESSING OF FORTRAN STATEMENTS " 1965	MELBOURNE, A & PUGHIR	54
DECISION SYSTEMS, INC " INTERIM STATUS REPORT NO. 1 " 1968		24
PLEMENTATIONS " 1964	STEVENS, W " THE STRUCTURE OF SYSTEM/360: PART II - SYSTEM IM	70
LIVERANCE, M " A SYSTEM/360 PROFILE -- READ-ONLY STORAGE " 1965		46
IME-SHARED PROCESSOR " 1968	HELLERMAN, L ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A T	37
IME-SHARED PROCESSOR " 1968	HOERNES, G ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A T	37
ANDAH, L " MICROPROGRAMMING AND STORED LOGIC " 1964		3
SEMARNE, M & PORTER, R " A STORED LOGIC COMPUTER " 1961		67
SEMARNE, M & MCGEE, W " STORED LOGIC COMPUTING " 1961		66
HILL, R " STORED LOGIC PROGRAMMING AND APPLICATIONS " 1964		30
LAWSON, M " PROGRAMMING-LANGUAGE-ORIENTED INSTRUCTION STREAMS " 1968		44
1953	STRINGER, J " MICROPROGRAMMING AND THE CHOICE OF ORDER CODE " 1961	71
CIRCUITS IN AN ELECTRONIC DIGITAL COMPUTER " 1953	MILKES, M & PITKOWSKY, S " STRUCTURAL ASPECTS OF THE SYSTEM/360, MODEL 85 " 1968	84
AL-PURPOSE MICROPROGRAM-CONTROLLED COMPUTER WITH ELEMENTARY STRUCTURE " 1960	KAMPE, T " THE DESIGN OF A GENER	23
" MICROPROGRAMMING THE SPECTRA 70/35 " 1966	N A VARIABLE STRUCTURE DIGITAL COMPUTER " 1962	19
1964	STEVENS, W " THE STRUCTURE OF SYSTEM/360: PART II - SYSTEM IMPLEMENTATIONS " 1968	70
969	BAZERQUE, G; FERRIE, J & HUGOT, P " MICROMACHINE UNIVERSAL STRUCTURE STUDY ORIENTED TO COMPUTERS SIMULATION " 1968	6
JE, G; FERRIE, J & HUGOT, P " MICROMACHINE UNIVERSAL STRUCTURE STUDY ORIENTED TO COMPUTERS SIMULATION " 1968	BAZERQ	72
" THE GROWTH OF INTEREST IN MICROPROGRAMMING: A LITERATURE SURVEY " 1969	MILKES, M	6
TING, INC " INFORMATION OUTLINE: REMOTE-ACCESS TIME-SHARING SYSTEM " 1961	ALLEN-BABCOCK COMPU	83
: CARTER, W; CHAFFORD, W & NOWELL, G " THE CONTROLS AUTOMATION SYSTEM " 1965	BUCKINGHAM, B	1
REELTZ, M; LEVY, S; LINHARDT, R & MILLER, H " SYSTEM ARCHITECTURE FOR LARGE SCALE INTEGRATION " 1967		17
BASHKOV, T; SASSON, A & KRONFELD, A " SYSTEM DESIGN OF A FORTRAN MACHINE " 1967		8
VOLLBRECHT, J " MIDAS: MICROPROGRAMMING DESIGN AID SYSTEM FOR RCA SPECTRA 70/45 " 1968		5
STEVENS, W " THE STRUCTURE OF SYSTEM/360: PART II - SYSTEM IMPLEMENTATIONS " 1964		77
TUCKER, S " MICROPROGRAM CONTROL FOR SYSTEM/360 " 1967		70
ANSMAN, T & WIMACK, K " 1401 COMPATIBILITY FEATURE ON THE IBM SYSTEM/360 MODEL 30 " 1965	MCCORMACK, M; SCH	73
WEBER, H " A MICROPROGRAMMED IMPLEMENTATION OF EULER ON IBM SYSTEM/360 MODEL 30 " 1967		49
STEVENSON, W " THE STRUCTURE OF SYSTEM/360: PART II -- READ-ONLY STORAGE " 1965		78
CONTI, C; GIBSON, D & PITKOWSKY, S " STRUCTURAL ASPECTS OF THE SYSTEM/360, MODEL 85 " 1968		46
GRAMMING AS A TECHNIQUE IN THE CONTROL OF DIGITAL COMPUTING SYSTEMS " 1958	MORRISON, G " INVESTIGATION OF MICRO-PRO	70
GERACE, G " MICROPROGRAMMED CONTROL FOR COMPUTING SYSTEMS " 1963		23
STRUGARU, C " A MICROPROGRAMMED UNIT FOR COMPUTER SYSTEMS " 1969		57
PATZER, W & VANDLING, G " SYSTEMS IMPLICATIONS OF MICROPROGRAMMING " 1967		28
DECISION SYSTEMS, INC " INTERIM STATUS REPORT NO. 1 " 1968		72
		60
		24
		24
		76
		57
34 ENTRIES WERE SUPPRESSED FOR KEYWORD -----> THE		
STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968	HOERNES, G ET AL " CONTROL	37
STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968	HELLERMAN, L ET AL " CONTROL	37
BABCOCK COMPUTING, INC " INFORMATION OUTLINE: REMOTE-ACCESS TIME-SHARING SYSTEM " 1961	ALLEN-	1
EMEL'YANOV-YARUSLAVSKY, L & TIMOFEEV, A " MICROPROGRAM CONTROL FOR DIGITAL COMPUTERS " 19		26

KWIC INDEX LISTING

DATE 10/23/69
TIME 01:42:03

INDEX

KEYWORD AND TEXT

2 ENTRIES WERE SUPPRESSED FOR KEYWORD -----> TO

SHITA,S " A MICROPROGRAMMED COMMUNICATION CONTROL UNIT, THE TOSBAC ON-231 " 1968 47
VAN DER POEL,W " MICRO-PROGRAMMING AND TRICKOLOGY " 1962 75
MC GEE,W " THE TRW-133 COMPUTER " 1964 50
TUCKER,S " MICROPROGRAM CONTROL FOR SYSTEM/360 " 1967 73
OREYER,L " PRINCIPLES OF A TWO-LEVEL MEMORY COMPUTER " 1968 25

STRUGARU,C " A MICROPROGRAMMED UNIT FOR COMPUTER SYSTEMS " 1969 72
MATSUSHITA,S " A MICROPROGRAMMED COMMUNICATION CONTROL UNIT, THE TOSBAC ON-231 " 1968 47
" THE DESIGN OF PROGRAM-MODIFIABLE MICRO-PROGRAMMED CONTROL UNITS " 1962 30
BAZERQUE,G; FERRIE,J & HUGOT,P " MICROMACHINE UNIVERSAL STRUCTURE STUDY ORIENTED TO COMPUTER< SIMULATION " 6
ILEVSKIY,P & KHOMYAKOV,M " A PROGRAMMING MICRO-LANGUAGE FOR URAL-4 " 1966 12
HOERNES,G ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHAR 37
ED PROCESSOR " 1968 37
HELLERMAN,L ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHAR 37

TRUL " 1967 74
VALASSIS,J " MODULAR COMPUTER DESIGN WITH PICOPROGRAMMED CON 74
VAN DER POEL,W " MICRO-PROGRAMMING AND TRICKOLOGY " 1962 75
VANDLING,G & WALDECKER,D " THE MICROPROGRAM CONTROL TECHNIQUE 76
PATZER,W & VANDLING,G " SYSTEMS IMPLICATIONS OF MICROPROGRAMMING " 1967 60
SOFTWARE INTERACTIONS: A MACHINE ORGANIZATION SOLUTION, THE VARIABLE INSTRUCTION COMPUTER " 1966SPENCE,A " HAROMARE AND 69
MILLER,E " RELIABILITY ASPECTS OF THE VARIABLE INSTRUCTION COMPUTER " 1967 56
EILSON,D " MICROPROGRAMMING THE SPECTRA 70/35 " 1966 CAMPBELL,C & N 19
OR URAL-4 " 1966 VASILIEVSKIY,P & KHOMYAKOV,M " A PROGRAMMING MICRO-LANGUAGE F 12
RCA SPECTRA 70/45 " 1968 BORSCHEV,V; VOLLBRECHT,J " MIOSAS: MICROPROGRAMMING DESIGN AID SYSTEM FOR 77

LOGIC DESIGN " 1969 76
VANDLING,G & WALDECKER,D " THE MICROPROGRAM CONTROL TECHNIQUE FOR DIGITAL 76
WILKE'S,M " THE BEST WAY TO DESIGN AN AUTOMATIC CALCULATING MACHINE " 1951 80
SYSTEM/360 MODEL 30 " 1967 WEBER,H " A MICROPROGRAMMED IMPLEMENTATION OF EULER ON IBM S 78
WEBER,H " MICROPROGRAMMING " 1969 79
GREENBERG,M & WEGBREIT,E " A MICROPROGRAMMED COMPUTER " 1968 32
HE CONTROL CIRCUITS IN AN ELECTRONIC DIGITAL COMPUTER " 1953WILKES,M & STRINGER,J " MICROPROGRAMMING AND THE DESIGN OF T 84
ACHINE " 1951 WILKES,M " MICROPROGRAMMING " 1959 81
WILKES,M " THE BEST WAY TO DESIGN AN AUTOMATIC CALCULATING M 80
WILKES,M " THE GROWTH OF INTEREST IN MICROPROGRAMMING " 1968 82
WILKES,M " THE GROWTH OF INTEREST IN MICROPROGRAMMING: A LIT 83

8 ENTRIES WERE SUPPRESSED FOR KEYWORD -----> WITH

MODEL 30 " 1965 MCCORMACK,M; SCHANSMAN,T & WOMACK,K " 1401 COMPATIBILITY FEATURE ON THE IBM SYSTEM/360 49
COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962 YAMAJCHI,H ET AL " THE KT PILOT COMPUTER - A MICROPROGRAMMED 33

DECISION SYSTEMS,INC " INTERIM STATUS REPORT NO. 1 " 1968 24
MCCORMACK,M; SCHANSMAN,T & WOMACK,K " 1401 COMPATIBILITY FEATURE ON THE IBM SYSTEM/360 MODEL 30 " 49
" THE BEST WAY TO DESIGN AN AUTOMATIC CALCULATING MACHINE " 1951 80
WILKES,M; TRINGER,J " MICROPROGRAMMING AND THE CHOICE OF ORDER CODE " 1953 71
OF THE CONTROL CIRCUITS IN AN ELECTRONIC DIGITAL COMPUTER " 1953WILKES,M & STRINGER,J " MICROPROGRAMMING AND THE DESIGN 84
BILLING,H & HOPMANN,W " MIKROPROGRAMM-STEUERWERK " 1955 10
GLANTZ,H " A NOTE ON MICROPROGRAMMING " 1956 29
MERCER,R " MICRO-PROGRAMMING " 1957 55
S A TECHNIQUE IN THE CONTROL OF DIGITAL COMPUTING SYSTEMS " 1958 57
WILKES,M " MICROPROGRAMMING " 1959 81
MORRISON,G " INVESTIGATION OF MICRO-PROGRAMMING A

KWIC INDEX LISTING

DATE 10/23/69
TIME 01:42:03

INDEX

KEYWORD AND TEXT

TEXT

BLANKENBAKER, J " LOGICALLY MICRO-PROGRAMMED COMPUTERS " 1960	11
CK-PROGRAM-CONTROLLED COMPUTER WITH ELEMENTARY STRUCTURE " 1960	42
OPLER, A " NEW DIRECTIONS IN SOFTWARE 1960-1966 " 1966	59
CONROY, E " MICROPROGRAMMING " 1961	21
CONROY, E & MEADE, R " A MICROINSTRUCTION SYSTEM " 1961	22
SEARNE, H & PORTER, R " A STORED LOGIC COMPUTER " 1961	67
SEARNE, H & MCGEE, W " STORED LOGIC COMPUTING " 1961	66
IVerson, K " A PROGRAMMING LANGUAGE " 1962	41
VAN DER POEL, W " MICRO-PROGRAMMING AND TRICKOLOGY " 1962	75
SIGN OF PROGRAM-MODIFIABLE MICRO-PROGRAMMED CONTROL UNITS " 1962	30
& TIMOFEEV, A " MICROPROGRAM CONTROL FOR DIGITAL COMPUTERS " 1962	26
ROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962	33
ROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962	33
A 70/35 " 1966 N A VARIABLE STRUCTURE DIGITAL COMPUTER " 1962	19
ROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962	33
GEPAE, G " MICROPROGRAMMED CONTROL FOR COMPUTING SYSTEMS " 1963	28
SCHLAEPPI, H " MICROPROGRAMMED COMPUTER CONTROL " 1963	65
HARRAND, Y " EVOLUTION OF MICROPROGRAMMING CONCEPTS " 1963	34
CAL ORGANIZATION OF THE PB 440 MICROPROGRAMMABLE COMPUTER " 1963	14
CONOMICAL MULTIPROGRAM COMPUTER WITH MICROPROGRAM CONTROL " 1963	2
CONOMICAL MULTIPROGRAM COMPUTER WITH MICROPROGRAM CONTROL " 1963	2
HILL, R " STORED LOGIC PROGRAMMING AND APPLICATIONS " 1964	38
BOUTWELL, E " THE PB 440 COMPUTER " 1964	13
BECK, L & KEELER, F " THE C-8401 DATA PROCESSOR " 1964	7
MCGEE, W " THE TRW-133 COMPUTER " 1964	50
ANDAH, L " MICROPROGRAMMING AND STORED LOGIC " 1964	3
CHARACTOR COMPUTER FOR HIGH LEVEL LANGUAGE INTERPRETATION " 1964	53
STRUCTURE OF SYSTEM/360; PART II - SYSTEM IMPLEMENTATIONS " 1964	70
BRILEY, B " PRIOR PROGRAMMING " 1965	16
BENJAMIN, R " THE SPECTRA 70/45 EMULATOR FOR THE RCA 301 " 1965	9
LIVERANCE, H " A SYSTEM/360 PROFILE -- READ-ONLY STORAGE " 1965	46
LEVY, L " STATE OF THE ART OF MICROPROGRAMMING " 1965	45
SON, H " MICROPROGRAMMING FOR DATA ACQUISITION AND CONTROL " 1965	52
ON, H " MICROPROGRAM CONTROL FOR THE EXPERIMENTAL SCIENCES " 1965	51
W; CRAWFORD, W & NOWELL, G " THE CONTROLS AUTOMATION SYSTEM " 1965	17
COMPUTER FOR THE DIRECT PROCESSING OF FORTRAN STATEMENTS " 1965	54
I401 COMPATIBILITY FEATURE ON THE IBM SYSTEM/360 MODEL 30 " 1965	49
HECKELMAN, R " SELF-CHECKING REDUNDANT MICROPROGRAMS " 1966	36
OPLER, A " NEW DIRECTIONS IN SOFTWARE 1960-1966 " 1966	59
J " MICROPROGRAMMING, EMULATORS AND PROGRAMMING LANGUAGES " 1966	31
ICAL DESIGN OF A MICROPROGRAMMED SPECIAL-PURPOSE COMPUTER " 1966	40
P & KHDYAKOV, M " A PROGRAMMING MICRO-LANGUAGE FOR URAL-4 " 1966	12
MPBELL, C & NEILSON, D " MICROPROGRAMMING THE SPECTRA 70/35 " 1966	19
ORGANIZATION SOLUTION, THE VARIABLE INSTRUCTION COMPUTER " 1966	69
OPLER, A " FOURTH-GENERATION SOFTWARE: THE REALIGNMENT " 1967	58
BRUEVER, M " ADAPTIVE COMPUTERS " 1967	15
TUCKER, S " MICROPROGRAM CONTROL FOR SYSTEM/360 " 1967	73
SLOBODYANYUK, T " A METHOD OF MINIMIZING MICROPROGRAMS " 1967	68
FLYNN, M & MACLAREN, M " MICROPROGRAMMING REVISED " 1967	27
AILEY, S " FASTER COMPUTER CONTROL WITH A READ-ONLY MEMORY " 1967	4
W & VANDLING, G " SYSTEMS IMPLICATIONS OF MICROPROGRAMMING " 1967	60
S, J " MODULAR COMPUTER DESIGN WITH MICROPROGRAMMED CONTROL " 1967	74
RELIABILITY ASPECTS OF THE VARIABLE INSTRUCTION COMPUTER " 1967	56
KAMPE, T " THE DESIGN OF A GENERAL-PURPOSE M	
GRASSELLI, A " THE DE	
EMELYANOV-YAROSLAVSKY, L	
YAMAUCHI, H ET AL " THE KT PILOT COMPUTER - A MIC	
MATSUSHITA, S ET AL " THE KT PILOT COMPUTER - A MIC	
CAMPBELL, C & NEILSON, D " MICROPROGRAMMING THE SPECTR	
HAGIWARA, H; AHO, K ET AL " THE KT PILOT COMPUTER - A MIC	
BOUTWELL, E & HOSKINSON, E " THE LOGI	
ROSE, G; SANDERSON, J ET AL " CIRRUSS, E	
ALLEN, M; PEARCEY, T; PENNY, J ET AL " CIRRUSS, E	
MEGGITT, J " A	
STEVENS, W " THE	
MCGEE, W & PETER	
MCGEE, W & PETERS	
BUCKINGHAM, B; CARTER,	
MELBOURNE, A & PUGHIRE, J " A SMALL	
MCCORMACK, M; SCHANSMAN, T & WOMACK, K "	
GREEN,	
INCALLS, R " LOG	
BORSCHKEV, V; VASILEVSKIY,	
N A VARIABLE STRUCTURE DIGITAL COMPUTER " 1962 CA	
N A HARDWARE AND SOFTWARE INTERACTIONS; A MACHINE	
PATZER,	
VALASSI	
MILLER, E "	

KWIC INDEX LISTING

1D/23/69
01:42:03

DATE
TIME

INDEX

KEYWORD AND TEXT

TEXT

ASSON,A & KRONFELD,A " SYSTEM DESIGN OF A FORTRAN MACHINE " 1967	BASHKOW,T: S	5
GRAPHIC, " A MICROPROGRAMMED GRAPHICAL-INTERFACE COMPUTER " 1967	ROSE,G " INTE	63
I " MICROPROGRAMMED CONTROL IN PROBLEM-ORIENTED LANGUAGES " 1967	HAWRYSKIENYCZ,	35
RAMMED IMPLEMENTATION OF EULER ON IBM SYSTEM/360 MODEL 30 " 1967	WEBER,H " A MICROPROG	78
ILLER,H " SYSTEM ARCHITECTURE FOR LARGE SCALE INTEGRATION " 1967	8EELITZ,H: LEVY,S: LINHART,R & M	8
DECISION SYSTEMS,INC " INTERIM STATUS REPORT NO. 1 " 1968		24
DREYER,L " PRINCIPLES OF A TWO-LEVEL MEMORY COMPUTER " 1968		25
GREENBERG,M & WEGBREIT,E " A MICROPROGRAMMED COMPUTER " 1968		32
WILKES,M " THE GROWTH OF INTEREST IN MICROPROGRAMMING " 1968		82
OSSEN,S " HARDWARE DESIGN REFLECTING SOFTWARE REQUIREMENTS " 1968		64
BOULOT,M " EXTRACTEUR MICROPROGRAMME D' INFORMATION RADAR " 1968		62
SON,H " PROGRAMMING-LANGUAGE-ORIENTED INSTRUCTION STREAMS " 1968		44
ARTZ,S " MODEL 4200 - 8200 READ-ONLY MEMORY CONTROL LOGIC " 1968		43
R-ORDER LANGUAGE FOR DESCRIBING MICROPROGRAMMED COMPUTERS " 1968		2D
OPROGRAMMED COMMUNICATION CONTROL UNIT, THE TOSBAC DN-231 " 1968		47
MICROPROGRAMMING DESIGN AID SYSTEM FOR RCA SPECTRA 70/45 " 1968		77
KOWSKY,S " STRUCTURAL ASPECTS OF THE SYSTEM/360, MODEL 85 " 1968		23
UNIVERSAL STRUCTURE STUDY ORIENTED TO COMPUTERS SIMULATION " 1968		6
TING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968		37
TING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968		37
STRUGARU,C " A MICROPROGRAMMED UNIT FOR COMPUTER SYSTEMS " 1969		72
MCCLURE,R & DAVIS,R " A SMALL MICROPROGRAMMED COMPUTER " 1969		48
CALHOUN,R " DIAGNOSTICS AT THE MICROPROGRAMMING LEVEL " 1969		18
WEBER,H " MICROPROGRAMMING " 1969		79
E COMPUTER-WITHIN-A-COMPUTER A FOURTH GENERATION CONCEPT " 1969		61
OWTH OF INTEREST IN MICROPROGRAMMING: A LITERATURE SURVEY " 1969		83
E MICROPROGRAM CONTROL TECHNIQUE FOR DIGITAL LOGIC DESIGN " 1969		76
K " 1401 COMPATIBILITY FEATURE ON THE IBM SYSTEM/360 MODEL 30 " 1965		49
OPROGRAMMED IMPLEMENTATION OF EULER ON IBM SYSTEM/360 MODEL 30 " 1967		78
BENJAMIN,R " THE SPECTRA 70/45 EMULATOR FOR THE RCA 301 " 1965		9
KLEIN,S & SCHWARTZ,S " MODEL 4200 - 8200 READ-ONLY MEMORY CONTROL LOGIC " 1968		43
8OUTWELL,E " THE P8 440 COMPUTER " 1964		13
JTWELL,E & HOSKINSON,E " THE LOGICAL ORGANIZATION OF THE P8 440 MICROPROGRAMMABLE COMPUTER " 1963		14
62 CAMPBELL,C & NEILSON,D " MICROPROGRAMMING THE SPECTRA 70/35 " 1966		19
" MIDAS: MICROPROGRAMMING DESIGN AID SYSTEM FOR RCA SPECTRA 70/45 " 1968		77
BENJAMIN,R " THE SPECTRA 70/45 EMULATOR FOR THE RCA 301 " 1965		9
KLEIN,S & SCHWARTZ,S " MODEL 4200 - 8200 READ-ONLY MEMORY CONTROL LOGIC " 1968		43
& PITKOWSKY,S " STRUCTURAL ASPECTS OF THE SYSTEM/360, MODEL 85 " 1968		23
	CONTI,C: GIBSON,D	
	BAZERQUE,G: FERRIE,J & HUGOT,P " MICROMACHINE U	
	HOERNES,G ET AL " CONTROL STORAGE USE IN IMPLEMEN	
	HELLERMAN,L ET AL " CONTROL STORAGE USE IN IMPLEMEN	
	CONTI,C: GIBSON,D & PIT	
	VOLLBRECHT,J " MIDAS:	
	MATSUSHITA,S " A MICR	
	CHU,Y " A HIGH	
	KLEIN,S & SCHW	
	RE	
	LAW	
	WEBER,H " A MICR	
	WILKES,M " THE GR	
	RAKOCZI,L " TH	
	VANDLING,G & WALDECKER,D " TH	
	MCCORMACK,M: SCHANSHAN,T & WOMACK	
	WEBER,H " A MICR	

DATE 10/23/69
TIME 01:42:03

INDEX

KEYWORD INDEX

- 1 ALLEN-BABCOCK COMPUTING, INC " INFORMATION OUTLINE: REMOTE-ACCESS TIME-SHARING SYSTEM "
- 2 ROSE, G; SANDERSUN, J ET AL " CIRBUS, ECONOMICAL MULTIPROGRAM COMPUTER WITH MICROPROGRAM CONTROL " 1963
- 3 ALLEN, J; PEARLEY, T; PENNY, J ET AL " CIRBUS, ECONOMICAL MULTIPROGRAM COMPUTER WITH MICROPROGRAM CONTROL " 1963
- 4 AMOHL, L " MICROPROGRAMMING AND STORED LOGIC " 1964
- 5 BAILEY, S " FASTER COMPUTER CONTROL WITH A READ-ONLY MEMORY " 1967
- 6 BASHKOW, T; SASSON, A & KRONFELD, A " SYSTEM DESIGN OF A FORTRAN MACHINE " 1967
- 7 BAZERQUE, G; FERRIE, J & HUGOT, P " MICROMACHINE UNIVERSAL STRUCTURE STUDY ORIENTED TO COMPUTERS SIMULATION " 1968
- 8 BECK, L & KEELER, F " THE C-8401 DATA PROCESSOR " 1964
- 9 BEELITZ, H; LEVY, S; LINHART, R & MILLER, H " SYSTEM ARCHITECTURE FOR LARGE SCALE INTEGRATION " 1967
- 10 BENJAMIN, R " THE SPECTRA 70/45 EMULATOR FOR THE RCA 301 " 1965
- 11 BILLING, H & HOPMANN, W " MIKROPROGRAMM-STEUERWERK " 1955
- 12 BLANKENBAKER, J " LOGICALLY MICRO-PROGRAMMED COMPUTERS " 1960
- 13 BORSCHEV, V; VASILEVSKIY, P & KHOMYAKOV, M " A PROGRAMMING MICRO-LANGUAGE FOR URAL-4 " 1966
- 14 BOUTWELL, E " THE PB 440 COMPUTER " 1964
- 15 BOUTWELL, E & HOSKINSON, E " THE LOGICAL ORGANIZATION OF THE PB 440 MICROPROGRAMMABLE COMPUTER " 1963
- 16 BREUER, M " ADAPTIVE COMPUTERS " 1967
- 17 BRILEY, B " PRIOR PROGRAMMING " 1965
- 18 BUCKINGHAM, B; CARTER, W; CRAWFORD, W & NOWELL, G " THE CONTROLS AUTOMATION SYSTEM " 1965
- 19 CALHOUN, R " DIAGNOSTICS AT THE MICROPROGRAMMING LEVEL " 1969
- 20 CAMPBELL, C & NEILSON, D " MICROPROGRAMMING THE SPECTRA 70/35 " 1966 " A VARIABLE STRUCTURE DIGITAL COMPUTER " 1962
- 21 CHU, Y " A HIGHER-ORDER LANGUAGE FOR DESCRIBING MICROPROGRAMMED COMPUTERS " 1968
- 22 CONROY, E " MICROPROGRAMMING " 1961
- 23 CONTI, C; GIBSON, O & PITKOWSKY, S " STRUCTURAL ASPECTS OF THE SYSTEM/360, MODEL 85 " 1968
- 24 DECISION SYSTEMS, INC " INTERIM STATUS REPORT NO. 1 " 1968
- 25 DREYER, L " PRINCIPLES OF A TWO-LEVEL MEMORY COMPUTER " 1968
- 26 EMELYANOV-YAROSLAVSKY, L & TIMOFEEV, A " MICROPROGRAM CONTROL FOR DIGITAL COMPUTERS " 1962
- 27 FLYNN, M & MACLAREN, M " MICROPROGRAMMING REVISITED " 1967
- 28 GERACE, G " MICROPROGRAMMED CONTROL FOR COMPUTING SYSTEMS " 1963
- 29 GLANTZ, H " A NOTE ON MICROPROGRAMMING " 1956
- 30 GRASSELLI, A " THE DESIGN OF PROGRAM-MODIFIABLE MICRO-PROGRAMMED CONTROL UNITS " 1962
- 31 GREEN, J " MICROPROGRAMMING, EMULATORS AND PROGRAMMING LANGUAGES " 1966
- 32 GREENBERG, M & WEGBREIT, E " A MICROPROGRAMMED COMPUTER " 1968
- 33 MATSUSHITA, S ET AL " THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962
- 34 MATSUSHITA, S ET AL " THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962
- 35 HAGIMARA, H; ANDO, K ET AL " THE KT PILOT COMPUTER - A MICROPROGRAMMED COMPUTER WITH A PHOTOTRANSISTOR FIXED MEMORY " 1962
- 36 HARRAND, Y " EVOLUTION OF MICROPROGRAMMING CONCEPTS " 1963
- 37 HARRYSZKIEWICZ, I " MICROPROGRAMMED CONTROL IN PROBLEM-ORIENTED LANGUAGES " 1967
- 38 HECKELMAN, R " SELF-CHECKING REDUNDANT MICROPROGRAMS " 1966
- 39 HUERNES, G ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968
- 40 HELLEMAN, L ET AL " CONTROL STORAGE USE IN IMPLEMENTING AN ASSOCIATIVE PROCESSOR FOR A TIME-SHARED PROCESSOR " 1968
- 41 HILL, R " STORED LOGIC PROGRAMMING AND APPLICATIONS " 1964
- 42 IBM " DATA PROCESSING: MICROPROGRAMMING PI COURSE "
- 43 INGALLS, R " LOGICAL DESIGN OF A MICROPROGRAMMED SPECIAL-PURPOSE COMPUTER " 1966
- 44 IVERSON, K " A PROGRAMMING LANGUAGE " 1962
- 45 KAMPEIT " THE DESIGN OF A GENERAL-PURPOSE MICROPROGRAM-CONTROLLED COMPUTER WITH ELEMENTARY STRUCTURE " 1960
- 46 KLEIN, S & SCHWARTZ, S " MODEL 4200 - 8200 READ-ONLY MEMORY CONTROL LOGIC " 1968
- 47 LAWSON, H " PROGRAMMING-LANGUAGE-ORIENTED INSTRUCTION STREAMS " 1968
- 48 LEVY, L " STATE OF THE ART OF MICROPROGRAMMING " 1965
- 49 LIVERANCE, H " A SYSTEM/360 PROFILE -- READ-ONLY STORAGE " 1965
- 50 MATSUSHITA, S " A MICROPROGRAMMED COMMUNICATION CONTROL UNIT, THE TOSBAC DN-231 " 1968
- 51 MCCLEURE, R & DAVIS, R " A SMALL MICROPROGRAMMED COMPUTER " 1969
- 52 MCCORMACK, M; SCHANSHAN, T & WOMACK, K " 1401 COMPATIBILITY FEATURE ON THE IBM SYSTEM/360 MODEL 30 " 1965

KWIC TEXT LISTING

DATE 10/23/69
TIME 01:42:03

INDEX-----KEYWORD INDEX-----

50 MCGEE,M " THE TRW-133 COMPUTER " 1964
51 MCGEE,M & PETERSON,H " MICROPROGRAM CONTROL FOR THE EXPERIMENTAL SCIENCES " 1965
52 MCGEE,M & PETERSON,H " MICROPROGRAMMING FOR DATA ACQUISITION AND CONTROL " 1965
53 MEGGITT,J " A CHARACTER COMPUTER FOR HIGH LEVEL LANGUAGE INTERPRETATION " 1964
54 MELBOURNE,A & PUGAIRE,J " A SMALL COMPUTER FOR THE DIRECT PROCESSING OF FORTRAN STATEMENTS " 1965
55 MERCER,R " MICRO-PROGRAMMING " 1957
56 MILLER,E " RELIABILITY ASPECTS OF THE VARIABLE INSTRUCTION COMPUTER " 1967
57 MORRISON,G " INVESTIGATION OF MICRO-PROGRAMMING AS A TECHNIQUE IN THE CONTROL OF DIGITAL COMPUTING SYSTEMS " 1958
58 OPLER,A " FOURTH-GENERATION SOFTWARE: THE REALIGNMENT " 1967
59 OPLER,A " NEW DIRECTIONS IN SOFTWARE: 1960-1966 " 1966
60 PATZER,W & VANDLING,G " SYSTEMS IMPLICATIONS OF MICROPROGRAMMING " 1967
61 RAKOCZI,L " THE COMPUTER-WITHIN-A-COMPUTER " A FOURTH GENERATION CONCEPT " 1969
62 REBOULOT,M " EXTRACTEUR MICROPROGRAMME D' INFORMATION RADAR " 1968
63 ROSE,G " INTERGRAPHIC " A MICROPROGRAMMED GRAPHICAL-INTERFACE COMPUTER " 1967
64 ROSEN,S " HARDWARE DESIGN REFLECTING SOFTWARE REQUIREMENTS " 1968
65 SCHLAEPPI,H " MICROPROGRAMMED COMPUTER CONTROL " 1963
66 SEMARNE,H & MCGEE,M " STORED LOGIC COMPUTING " 1961
67 SEMARNE,H & PORTER,R " A STORED LOGIC COMPUTER " 1961
68 SLOBODYANYUK,T " A METHOD OF MINIMIZING MICROPROGRAMS " 1967
69 SPENCE,A " HARDWARE AND SOFTWARE INTERACTIONS; A MACHINE ORGANIZATION SOLUTION, THE VARIABLE INSTRUCTION COMPUTER " 1966
70 STEVENS,M " THE STRUCTURE OF SYSTEM/360; PART II - SYSTEM IMPLEMENTATIONS " 1964
71 STRINGER,J " MICROPROGRAMMING AND THE CHOICE OF ORDER CODE " 1953
72 STRUGARU,C " A MICROPROGRAMMED UNIT FOR COMPUTER SYSTEMS " 1969
73 TUCKER,S " MICROPROGRAM CONTROL FOR SYSTEM/360 " 1967
74 VALASSIS,J " MODULAR COMPUTER DESIGN WITH MICROPROGRAMMED CONTROL " 1967
75 VAN DER POEL,W " MICRO-PROGRAMMING AND TRICKLOGY " 1962
76 VANDLING,G & WALDECKER,D " THE MICROPROGRAM CONTROL TECHNIQUE FOR DIGITAL LOGIC DESIGN " 1969
77 VOLLBRECHT,J " MIDAS: MICROPROGRAMMING DESIGN AID SYSTEM FOR RCA SPECTRA 70/45 " 1968
78 WEBER,H " A MICROPROGRAMMED IMPLEMENTATION OF EULER ON IBM SYSTEM/360 MODEL 30 " 1967
79 WEBER,H " MICROPROGRAMMING " 1969
80 WILKES,M " THE BEST WAY TO DESIGN AN AUTOMATIC CALCULATING MACHINE " 1951
81 WILKES,M " MICROPROGRAMMING " 1959
82 WILKES,M " THE GROWTH OF INTEREST IN MICROPROGRAMMING " 1968
83 WILKES,M " THE GROWTH OF INTEREST IN MICROPROGRAMMING: A LITERATURE SURVEY " 1969
84 WILKES,M & STRINGER,J " MICROPROGRAMMING AND THE DESIGN OF THE CONTROL CIRCUITS IN AN ELECTRONIC DIGITAL COMPUTER " 1953

KWIC KEY WORD SUPPRESSION LISTING

10/23/69
01:42:03

DATE
TIME

KEY WORDS SELECTED FOR SUPPRESSION-----

AL
AND
ET
IN
OF
ON
THE
TO
WITH

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) The MITRE Corporation Bedford, Massachusetts		2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	
		2b. GROUP	
3. REPORT TITLE ANNOTATED MICROPROGRAMMING BIBLIOGRAPHY			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) N/A			
5. AUTHOR(S) (First name, middle initial, last name) Microprogramming Group (Thomas P. Berschback, coordinator)			
6. REPORT DATE JULY 1970		7a. TOTAL NO. OF PAGES 70	7b. NO. OF REFS 0
8a. CONTRACT OR GRANT NO. F19(628)-68-C-0365		9a. ORIGINATOR'S REPORT NUMBER(S) ESD-TR-70-204	
b. PROJECT NO. 700A			
c.		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) M69-65	
d.			
10. DISTRIBUTION STATEMENT This document has been approved for public release and sale; its distribution is unlimited.			
11. SUPPLEMENTARY NOTES N/A		12. SPONSORING MILITARY ACTIVITY Directorate of Planning and Technology, Electronic Systems Division, AF Systems Command, L. G. Hanscom Field, Bedford, Massachusetts	
13. ABSTRACT This bibliography provides an indexed, annotated list of publications related to the field of microprogramming, from the introduction of the term in 1951 to the present time.			

14.	KEY WORDS	LINK A		LINK B		LINK C	
		ROLE	WT	ROLE	WT	ROLE	WT
	MICROPROGRAMMING						
	MICROPROGRAMMING BIBLIOGRAPHY						
	ANNOTATED MICROPROGRAMMING BIBLIOGRAPHY						

DEPARTMENT OF THE AIR FORCE
HEADQUARTERS ELECTRONIC SYSTEMS DIVISION (AFSC)
LAURENCE G. HANSCOM FIELD, BEDFORD, MASSACHUSETTS 01730



REPLY TO
ATTN OF:

~~ESAI~~ ESGEI/5322

16 June 1970

SUBJECT: Review of Technical Document

TO: ESTI

Identification: ESD-TR-70-204

ESAI has no objection to release of the above-cited document to the Clearinghouse for Scientific and Technical Information.

John T. O'Brien
JOHN T. O'BRIEN
Chief, Public Information Division
Information Office

1 Atch
n/c